DTU Library

Continuous Platform Development
Synchronizing Platform and Product Development

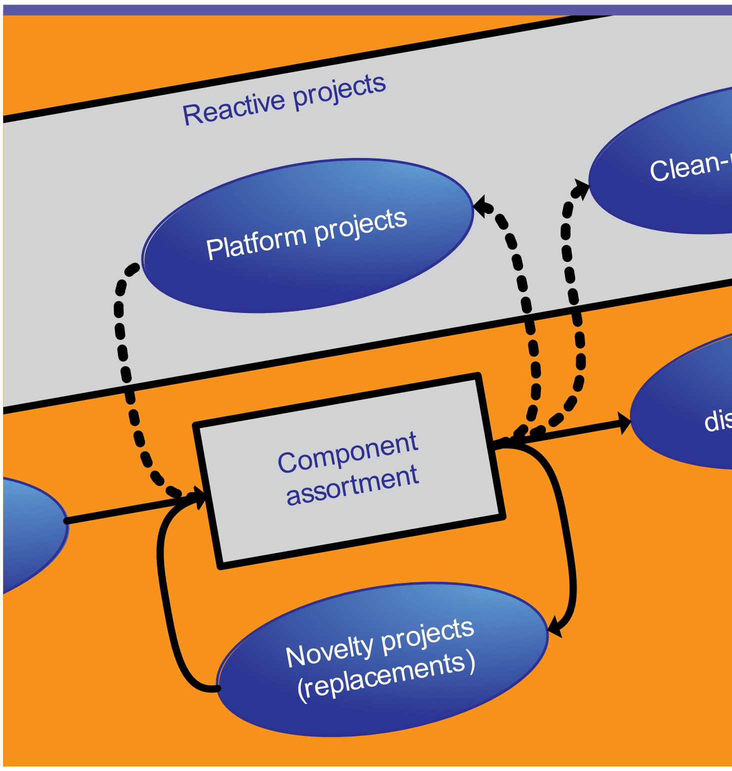
Nielsen, Ole Fiil

Publication date:
2010

Document Version
Publisher's PDF, also known as Version of record

Continuous Platform Development

– Synchronizing Platform and Product Development



PhD thesis 6.2010

DTU Management Engineering

Ole Fiil Nielsen

February 2010

DTU Management Engineering
Department of Management Engineering

Titel	Continuous Platform Development - Synchronizing Platform and Product Development
Type	Ph.d.-afhandling
Forfatters navn	Ole Fiil Nielsen
Årstal	2010
ISBN-nr.	978-87-90855-88-8
Institut navn	DTU Management Institut for Planlægning, Innovation og Ledelse Produktionstorvet 424 2800 Lyngby Tlf: 4525 4800 Mail: phd@man.dtu.dk
Tryk	Schultz Grafisk A/S

Continuous Platform Development

Synchronizing Platform and Product Development

A PhD dissertation submitted to
the Department of Management Engineering
of the Technical University of Denmark

by
Ole Fiil Nielsen
February 23rd 2009

Engineering Design and Product Development
Department of Management Engineering
DTU

This thesis is the result of a four year long process for me. The process has been long and sometimes tedious, but also extremely educating and full of experiences. I owe a lot to my colleagues at DTU and at LEGO, whom I have fortunately shared parts of this experience with.

I would like to thank Lone Munk, Rasmus Pedersen, and Morten Kvist, whom I have studied with as a PhD student, and with whom I have shared the ups and downs of our first venture into the research community.

I would also like to thank the employees at LEGO, whom I have also come to consider my colleagues, and who has treated me as a colleague and one with whom they could share their experiences with. Special thanks go to Henrik Priess Sørensen and Dorthe Kjærulff, who have taken me into their departments and with whom I have discussed my research and its appliance many times.

A very special thanks goes to Thomas Steen Jensen who have been my closest colleague and working partner at LEGO for the entire period. Most of the observations and findings presented in this thesis are as much his as they are mine, and while I have had the responsibility of ensuring the soundness of the reasoning and research approach and documentation, it has always been on his shoulders that responsibility of the economic consequences and implementation rested.

Another special thanks goes to my supervisor, Niels Henrik Mortensen – especially for his catching enthusiasm and eager interest for the research topic. He has set up the frame for this research and not only made this possible, but also interesting and fun.

Abstract

Platform-based product development is about sharing components and product architectures between products in a product family to save costs and free engineering resources. The intension is that the components and product architectures must be shared both across existing and future products, but continuous product family evolution challenges this strategy.

The concept of continuous platform development is based on the fact that platform development should not be a one-time experience but rather an ongoing process of developing new platforms and updating existing ones, so that product family evolution does not make the platforms irrelevant.

This research is based on case studies and applications in the two Danish companies, LEGO and Grundfos, where several platform development projects have been followed. LEGO is exceptional for their long experience with platform development and maintenance, while Grundfos is relatively new within this area.

This study puts emphasis on platform projects and defines such projects as something very different from product development projects and more similar to industrial research projects, due to the fact that they are continuous activities with relatively low risks and investments but also with relatively fuzzy results.

When looking for new platform projects, it is important to make sure that the company and market is ready for the introduction of platforms, and to make sure that people from marketing and sales, product development, and downstream departments are all consulted. Platform ideas originate primarily from experienced workers, managers, or platform thinkers.

Platform projects are presented regularly for future users and decision makers in structured presentations consisting of both quantitative estimates of costs and benefits and qualitative platform concept descriptions. This happens in a structured process, where platform projects are developed in parallel and prioritized in a stage-gate process.

When platform projects end the responsibility of the platform components is passed on to the platform maintenance organization. Product developers can add new platform components and clean-up and replacements of platform components are organized. Platform performance can be evaluated based on platform component usage and growth.

Platform development is effectively split from product development, but platform project are aligned through constant presentations and product developer involvement. When the two processes are brought back together timing of platform project is essential because it must be ready in time – too early starts can however make the platform irrelevant.

1. Introduction

This research is about platform development. Development of one or more platforms is the necessary prerequisite to conducting product development based on platforms (see Figure 1). The concepts of platform-based product development and platform development are not my own invention and have been described and documented through the last decades in numerous journal papers, books, and conference proceedings.

The term *continuous* platform development is my own invention (the term have another meaning in [Meyer 1997]). The responsibility of product developers of any company is not only to develop one new product, but rather to continuously develop new products and improve on existing products – continuously improving the product assortment. Likewise the responsibility of platform developers is not only the development of one singular platform, but rather to continuously develop new platforms and improve on existing platforms – continuously improving the platform assortment. With the addition of *continuous*, I hope to transform the traditional simple understanding of platforms as something very static into a more dynamic concept.

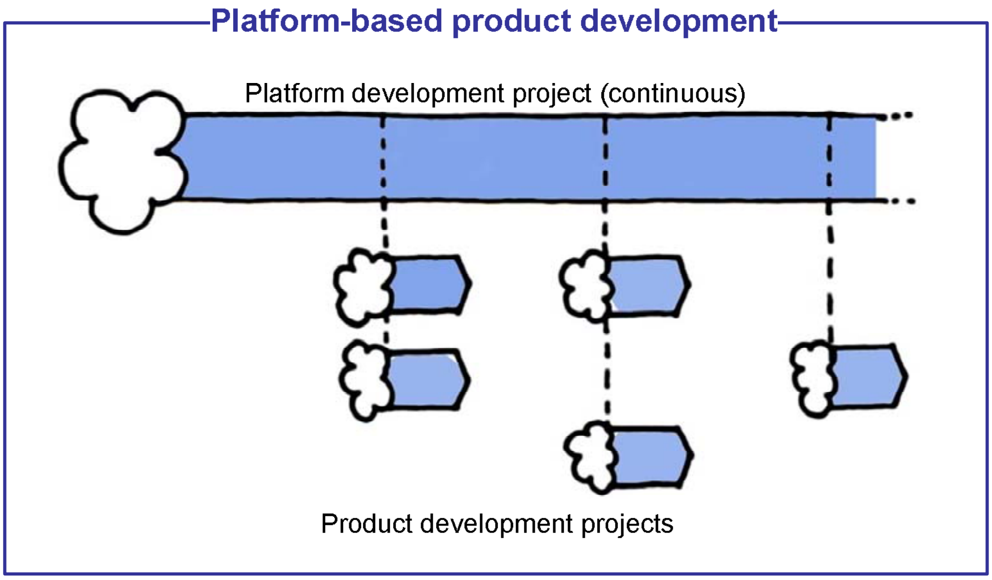


Figure 1: Platform-based product development is a scheme wherein product development projects are derived from a platform concept. This scheme is described more thoroughly throughout this thesis.

1.1. Background

Since research in platform development has mostly dealt with the development of the platform itself, the amount of literature describing the work tasks and responsibilities of platform developers is limited. This however is a problem in itself because the work tasks and responsibilities of platform developers are likely to influence the way they are able to conduct platform development. Like platform-based product development influences the way product development is conducted, so is continuous platform development likely to influence the way platform development is conducted.

Furthermore, platform developers looking for guidance on how to manage an assortment of different platforms – or simply how to keep a platform alive in an organization, have had few and vague answers from the research community. The aim of my research is to change this.

The following figure, Figure 2, summarizes this reasoning. A more detailed analysis of state-of-the-art research and industrial practice will be given later in this thesis.

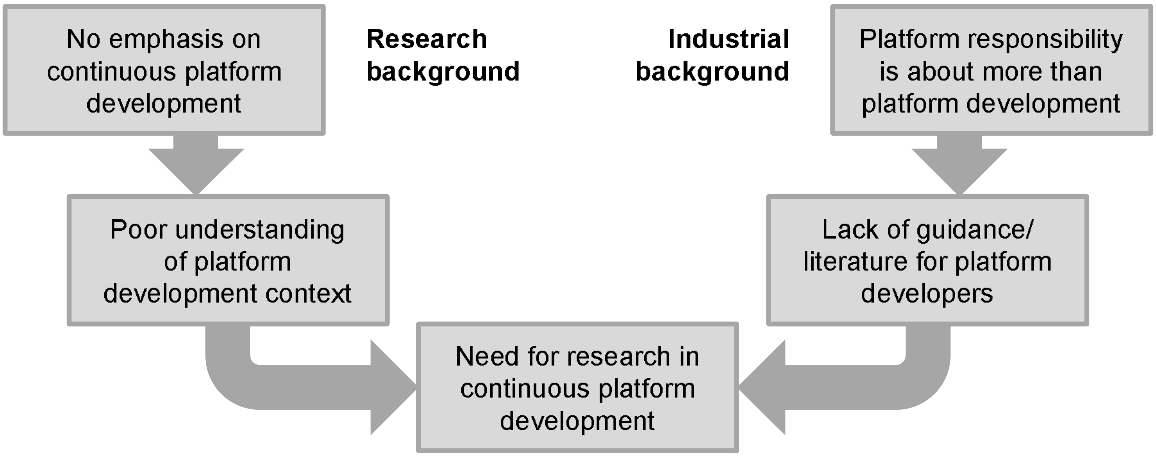


Figure 2: Reasoning behind this research project.

1.2. Research objects and aims

The primary aim of this research project is to improve the understanding of continuous platform development and how to conduct it. This improved understanding is intended for researchers as well as practitioners within the fields of product architecture, product platforms, product families, and related fields. Intended practitioners are Design Engineers, Product Managers, Project Managers, and R&D Managers, who are developing or managing multiple products. Researchers and practitioners from others areas of product development may also gain insights, especially from the descriptions on how platform development relates to and can be synchronized with product development projects.

It is also my aim with this research project to create several models summarizing and visualizing this acquired understanding, so that this understanding will be passed on to others more easily.

Gaining understanding of continuous platform development means researching what precedes platform development projects and what succeeds them, as well as researching how platform development projects interacts with simultaneous product development projects and other platform development projects.

To be more specific, my research objects are the processes of finding new platforms, improving on existing platforms, and terminating obsolete platforms. Special attention is given to decision points, where various platform projects are evaluated and prioritized.

The following figure (Figure 3) shows the research object, and how the study of this object should lead to a gained understanding of the object. Next, I will create models of this gained understanding (this will be the tangible result of the research), so that the readers of this thesis can gain an understanding of the research object (this is the research aim). Finally and hopefully – and this is not shown in the figure, readers should be able to better understand other/own observations of this research object using the mentioned models.

I have undertaken several case studies during this research project to study management of platform assortments. Most of the cases have taken place at LEGO Company, where a large number of minor platform projects have been undertaken over a period several years, this has created an unique opportunity to study how an assortment of platforms perform and how they influence each other and product development in general.

.

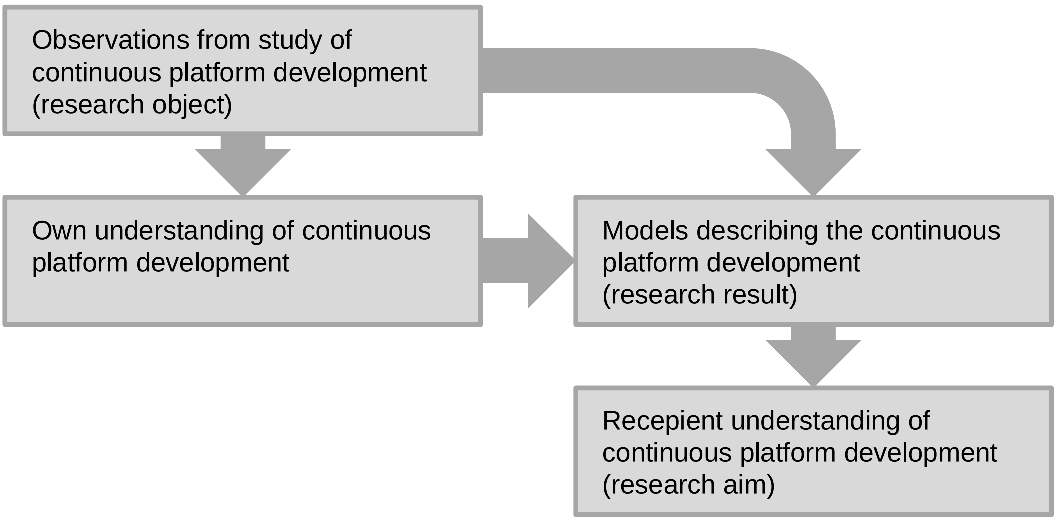


Figure 3: Relation between research object, research result, and research aim.

An additional case study has been undertaken at Grundfos, where I followed two specific platform projects. In this study I have had the opportunity to study how the understanding acquired in the LEGO case studies could be applied and implemented.

1.3. Research approach

The empirical studies in LEGO and Grundfos, which form the basis of this research project, have been undertaken using the *participatory action research* approach (Figure 4). The way this has been carried out is that I have joined teams in LEGO and Grundfos, which first planned the activities, then carried them out while observing and reflecting upon the consequences of the activities.

The reason for choosing participatory action research is two-fold: First, I could offer my assistance in the projects and therefore be a more attractive collaborator; Secondly, I could influence what activities were undertaken and therefore try out possibilities, which my non-researcher partners were not likely to undertake on their own (this is a also a commonly accepted reason for choosing participatory action research [Wadsworth 1998]).

The research project can be characterized in more detail using the framework offered in the *design research methodology* [Blessing et al. 2002] (see Figure 5). The detailed empirical studies in my cases form what is called *descriptive study I*, and this thesis, the understanding it represents, and the models which are presented in it form what is called *prescriptive study*. Finally some parts of the later case studies also form an initial *descriptive study II*, because some of the results from the *prescriptive study* were tested in these projects.

Participatory Action Research:

1. Involves all relevant parties in actively examining together current action in order to change and improve it.
2. Action which is researched, changed and re-researched, within the research process by participants.
3. Aims to be active co-research, by and for those to be helped.
4. Tries to be a genuinely democratic or non-coercive process whereby those to be helped, determine the purposes and outcomes of their own inquiry.

Figure 4: Participatory Action Research [Wadsworth 1998]

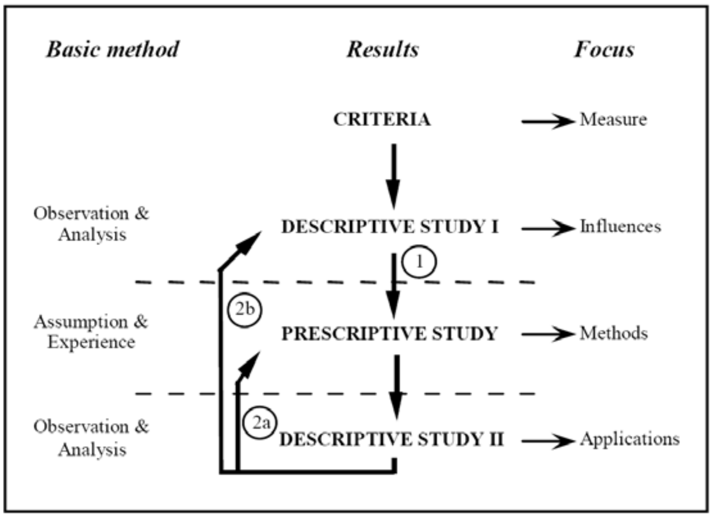


Figure 5: Design Research Methodology [Blessing et al. 2002]

Compared with the list of typical research projects in the *design research methodology* [Blessing et al. 2002], this research project has a review of the *criteria*, a detailed *descriptive study I*, a detailed *prescriptive study*, and a initial *descriptive study II*.

The consequence of focusing on the *descriptive study I* and the *prescriptive study* naturally leaves the results of this research largely unverified. However, as the results are based on the multiple cycles of planning, taking action, and observing (i.e. action research), they are not simple suggestions based on observations, but lessons learned through trial and error.

Moreover a detailed verification of results like those presented in this research project would require a large number of available cases, which is simply not present at this time.

I am therefore confident that I have chosen a research approach which matches well my research object and the aims of the project.

1.4. Research data

This research is based upon data collected using a number of data collection methods. Primarily the participatory action research method have been used, combined with follow-up interviews, direct observations, and analyses of documentation and physical artifacts, which are all well-documented data collection methods [Yin 1994]

A senior researcher at my university has classified research within platform-based product development as macro-research, because the researchers studying within this field observes and records phenomenon and data on a relatively high level. Many researchers within other areas will record interviews and videotape meetings and workshops. This kind of research project however involves new interviews and meetings on a weekly basis for several years and the raw data is scattered throughout countless presentations, posters, and notes, which is simply too extensive to present in any meaningful way.

The data that is presented in this thesis are therefore extracted from numerous sources as field notes [Wilkinson 1995], where each of these sources has only contributed very little to the whole. They are for the most part various conclusions and general observations, which have later been compiled into the presented case descriptions.

On the positive side, the field notes are first hand information, which has not gone through any interpreter, and which has been confirmed through multiple observations in various projects (sometimes referred to as triangulation) [Patton 1987].

The observations are often long-term observations on a high level. I am confident that other research approaches, which focused on more detailed studies, could therefore have given valuable results which I may have overlooked, but I am likewise confident that these approaches would then not have captured the overview and high-level insights that my research has resulted in.

The relatively high level of the observations also affects the validation of the results. On this high level numerous factors affect the outcome of projects and interfere with the applied methods, and it is very hard to observe the consequences of changing a method or procedure because other factors are not equal. Therefore most conclusions in this area of research need many case study confirmations to be valid and a few of the conclusions in this thesis are more indications than they are truly valid conclusions. I believe however that that is an unavoidable condition of doing research in this kind of research area rather than an indication of poor quality.

1.5. Research frame

My research is heavily founded on a long tradition of research in product development. Product development thus forms the overall discipline of my research. I describe product development as a discipline comparable to politics, theology, medicine, or other bodies of knowledge.

The broad nature of this research project, however, sometimes cross the boundary of traditional product development going slightly into other disciplines such as management and economics. It is important to note however, that even though the research also builds upon certain management and economic theories and even though the results may have implication on management and economics, the main focus remains within product development and it is also here that the scientific contributions lie.

Platform-based product development is a special kind of product development. In its simple form it can be described simply as a subcategory of product development, namely the one that deals with developing multiple products simultaneously. However, in the form it is used in this research it is more than this. Essentially all product development can be seen as degrees of platform-based product development, since all product development projects share some physical or non-physical elements and this commonality can be turned into an advantage.

In this version platform-based product development is not a subcategory, but instead a way of perceiving product development. The platform thinker perceives everything as being made up of unique and shared parts – and perceives all variety as being either valuable, when it differentiates one thing from another in a way so that each thing has its own relative strengths and weaknesses, or not-valuable, when there is no rational reason for the variety.

It may even be argued that platform thinking is broader than product development, because this differentiation between unique/shared and valuable/not-valuable can be applied in many other situations where an assortment of things is analyzed or created.

Because of this I will characterize platform-based product development as a paradigm. This paradigm is shared by a number of product development researchers, whereas other product development researchers perceive product development differently.

According to [Kuhn 1996] paradigms refer to a set of practices within a discipline, but where the paradigms in [Kuhn 1996] cannot co-exist and only apply to natural sciences, I use the term in the less strict version, which has evolved since his initial publication in 1962, in this version [Törnebohm 1974, Arbnor et al. 1997] paradigms are allowed to co-exist and do apply and are extensively used in the social sciences to which product development belong.

Since platform-based product development offer one view out of several on product development, it is important to note the limitations of research based on this paradigm. The results of this research project are found within platform-based product development and they apply within this paradigm. Making general conclusions on product development from the research presented here is one-sided as other aspects are not dealt with and therefore I primarily offer results within platform-based product development and only speculate how this influence product development in general.

1.6. Thesis disposition

The remaining thesis is build up in the following way:

- Section 2, The Platform Paradigm, describes the research frame namely the platform paradigm in more detail based on state-of-the-art literature and common understanding of the various concepts, which relate to this research area.
- Section 3, Research Questions and Hypotheses, then poses a set of research questions which become relevant when platform development is perceived as an ongoing continuous process.
- Section 4, Case Material, presents a number of cases which will be my primary source of inspiration, observation, and validation in the following sections. This section does not include raw data in the form of interviews and full documents, because this material is too extensive and not presentable in any form. Instead the section gives an overview of the cases, from which the observations originate.
- Section 5, Platform Projects, then extends the platform paradigm by describing and classifying various kinds of platform projects in order to set the stage for the following sections.

The following three sections answers research questions and constitute the scientific contribution of this thesis. The descriptions are however not limited to short fact-based argumentations, but try to give a comprehensive holistic description of the sub-area. Each section can be read independently and contain its own conclusions.

- Section 6, Finding and Screening Platform Projects, details the early stages of platform development and prescribes a structured process for initiating and developing platform projects.
- Section 7, Maintaining Platforms, describes how platform components are maintained, how uncontrolled growth is prevented and how platform performance can be evaluated based on component data.
- Section 8, Synchronizing Platform and Product Development, describes how platform development is split from product development, how the two processes run in parallel, and how they are finally brought back together.
- Section 9, Conclusion, finally summarizes the contributions and reflects on the validity and implications of the results.

The paper [Mortensen et al. 2009] has been appended for reference. This paper describes parts of the research and is referred to in this thesis.

2. The Platform Paradigm

In its core platform-based product development is about sharing:

The conscious sharing of *things* in product development between products in a product family in order to attain a number of *effects* in the product development stage itself or in the succeeding stages of a company's value chain (Figure 6).

- Typical *things* that are shared are *product components* and *product architectures*. However, it can also often be sharing of components and architectures not directly related to the products, like manufacturing equipment, processes and/or supply chain routes. In the successful cases of platform-based product development that my colleagues and I have studied, many different *things* are shared, and it is this combination of *things*, which I refer to as a *platform*.
- Typical *effects* that are attained are *reductions in order lead-time* and *reductions of product costs*. These effects are attained in specific stages of the company's value chain. Typical examples are reductions in project time and costs in product development, which is essential in many engineering-to-order setups, and/or reductions in production/assembly time and costs.

The specific *things* and *effects* differ from case to case. It is the tenet – and not the specific *things* and *effects*, which is common for all of these cases.

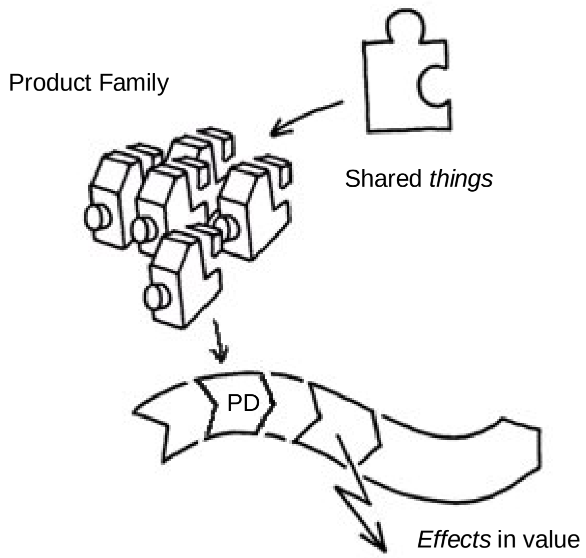


Figure 6: Platform-based product development - The conscious sharing of *things* in order to attain a number of *effects*.

2.1. Product platforms

A key part of platform-based product development is the product platform itself. Another key part is the product family. A product family is a set of products, which are related in some way. This relation between the products is what differentiates a product family from a traditional product portfolio [Sawhney 1998], and most often this is based on a shared product platform [Sahin et al. 2006].

Although many different authors have given their perception on what a product platform is through the last decades, most of these perceptions fall into the following three categories. The three perceptions are not mutually exclusive, and they are primarily three different view-points – not three different phenomena. Many platform cases can be described from several combinations of these view-points. When implementing a platform these view-points can however make a difference on the way individual platform components are managed.

To illustrate the differences and similarities I will show a small example of a product family (Figure 7) and the different platform concepts to which it corresponds.

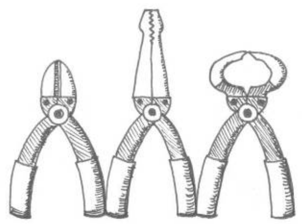


Figure 7: A product family (comparable to product platforms given in the next three figures).

2.1.1. Building block platforms

One of the original contributors to the platform paradigm [Meyer et al. 1997] perceives product platforms as a set of common components, modules, or parts from which a stream of derivative products can be efficiently created and launched. Or (in the same text) as a set of subsystems and interfaces that form a common structure from which a stream of derivative products can be efficiently developed and produced.

Similar to this perception is the one from [Robertson et al. 1998]. Here, the product platform is described as the collection of assets that are shared by a set of products. These assets can be divided into four categories: Components, Processes, Knowledge, and People and relationships. Taken together, these shared assets constitute the product platform. Generally, platform products share many if not most development and production assets.

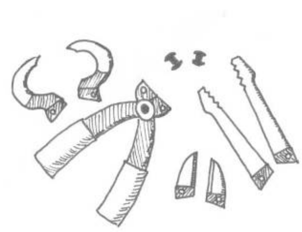


Figure 8: A building block platform is the sum of all available things, which can be used to create products.

The building block platform is the sum of all available common things, from which products can be created (Figure 8). Most authors use the word modules about the shared items of a platform. In this case the modules are standardized units, which are available for product building.

2.1.2. Cornerstone platforms

The platform perception given in [Blackenfelt 2001] is one, which seems very popular in the industry. Here the platform is only the common, necessary parts. A product family is seen as all the products that are based on the same platform (i.e. a product family is defined by the commonality in parts). Thus one platform results in one product family. This platform perception is also presented in [Muffatto 1999], where it is added that platform components are physically connected.

Likewise [Ulrich et al. 2000] describes the platform as a collection of assets, including component designs, which are shared by the products of a product family.

Figure 9: A cornerstone platform is the sum of all the things, which are shared between several products. It does therefore not include things that are only present in one product.

The cornerstone platform is the sum of those shared things, which are present in all completed product variants (See Figure 9). It is therefore much more focused and limited compared to the building block platform. If the platform consists of sub-things, these things are often called modules. They are standardized units, which are shared in all product variants.

2.1.3. Skeletal platforms

The final platform perception category is actually the most popular one in the state-of-the-art literature.

[Erens 1996] describes a product platform as an architectural concept comprising interface definitions and key-components, addressing a market and being a base for deriving different product families.

Likewise [Sanchez et al. 1999] describes how modular product architectures serve as efficient "platforms" for leveraging families of products to meet market demands for product variety more quickly and efficiently.

Finally [Meyer et al. 1997] also uses this description: "The combination of subsystems and interfaces defines the architecture of any single product. Every product has an architecture; the goal is to make that architecture common across many products. Any single product's architecture therefore has the potential to become a product platform architecture if it is designed and then used as the basis for creating several more derivative products."

I assume from this that [Meyer et al. 1997] perceives the platform as both a common architecture and a set of shared assets (see 2.1.1 Building block platforms). This is also the case in [Miller 2000], which state that the combination of reference architectures and shared assets is often called a platform.

To summarize these contributions, the architecture of a conceived generic product can be shared as a platform (Figure 10). When comparing with the other types of platforms, this platform type focuses less on the actual shared physical components of the products, and more on the shared architecture.

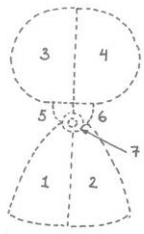


Figure 10: A skeletal platform is in fact a generic architecture, which is used as a platform for deriving product variants. Each of the bordered sections represents generic placeholder modules, which correspond to one or more interchangeable units.

The skeletal platform or architecture also consists of modules, although these modules are not standardized units. Instead they are generic placeholder units, meaning that they are a general abstraction, which correspond to one or more interchangeable units.

Each of these generic placeholder modules define the design rules, which all the corresponding interchangeable units must comply with.

2.1.4. Platforms in this thesis

Figure 7 though Figure 10 illustrate that all platform categories can be used to describe the same product family. The above descriptions of platforms are not necessarily preclusive, and several authors apply more than one of these descriptions. In this thesis I will therefore take one step back and focus on the sharing which is common for all three platform types. As stated in the beginning of section 2, I will describe the platform as the combination of things that are shared.

The above platform types are instead aspects of platforms, and any given platform in this thesis can be comprised of one or more of these aspects. This is especially true for the final platform aspect – the skeletal platform, since all sharing essentially requires some degree of common modular architecture.

2.2. Modular architectures

A typical way of introducing commonality (i.e. sharing) between products is to give the products *common modular architectures*.

Architectures for products are often defined as the arrangement of functional elements, the mapping from functional elements to physical components, and the specification of interfaces among interacting components [Ulrich 2007, Ulrich 1995].

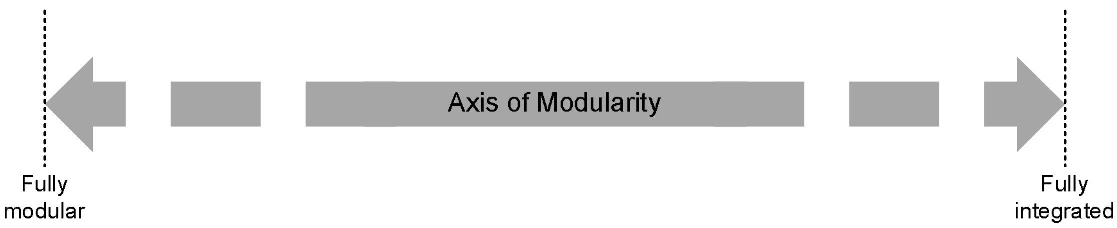


Figure 11: The axis of modularity can be used to describe the architecture of products [Ulrich et al. 2000].

Any product architecture can be described on an axis of modularity (Figure 11) [Ulrich et al. 2000]. This axis spans from fully modular in the one end to fully integrated in the opposite end.

Modular architectures are architectures where a product is divided into well-confined (i.e. independent) modules, which are connected to each other by well-defined interfaces. On the other hand integrated architectures are architectures where all components are intertwined and cannot be clearly defined. Changes in any component in a modular architecture can often be contained within one module, leaving the remaining product unchanged, whereas changes in any component in an integrated architecture are likely to affect the whole product [Ulrich et al. 2000].

Fully modular and fully integrated architectures are mostly abstractions, since modularity is a *relative* property [Ulrich et al. 1991]. Most product architectures have can be perceived as modular or integrated depending on the reference. The axis can however be used in comparisons. Product designers can make product architectures *more* modular or *more* integrated.

Optimizations on a single-product level such as efforts to reduce the costs or improve the quality of a particular product will often result in more integrated architectures. The flexibility of the modular architecture naturally is more costly and therefore needs to serve a purpose. A detailed review of the literature on modularity definitions, which also highlights some of the differences, can be found in [Gershenson et al. 2003].

There are many purposes for having modular architectures, some of which has to do with commonality and others which have not (e.g. for maintenance, for servicing, for updating, or for assembly). A comprehensive description of modularization and its different purposes is summarized in the module drivers proposed by [Erixon 1998].

When products are modularized with the purpose of introducing commonality, the components of the products are divided according to their use across the assortment of different products. This means that components, which will be used in the same patterns of products, are grouped together in modules, and components, which are used in different patterns, are divided into separate modules.

2.2.1. Common modular architectures

It is possible to share modular architectures between products, so that each of the products complies with the same architecture. This of course requires the architecture to be generic, meaning that the architecture consists of at least one and possibly many generic placeholder modules, which can be swapped or omitted in order to generate the individual product variants. This type of architecture is sometimes referred to as a *reference architecture* [Harlou 2006] and can be used as a platform for deriving product variants (i.e. a skeletal platform).

Generic modular architectures can also be the basis of a configuration system. A configuration system makes it possible to configure product variants based on a set of rules dictated by a common modular architecture [Hvam et al. 2008]. Some companies cannot introduce configuration systems, because their product families are not based on common generic modular architectures.

2.2.2. Individual modular architectures

Some products (like most LEGO products) have no *common* modular architecture but still have modular architectures. These architectures have no generic placeholder modules, but can instead use common modules (e.g. LEGO bricks). The architectures are still modular in the way that they are designed to incorporate standard components.

Some standard components like screws, bolts, wires and building or raw material-like components (e.g. metal sheets or rods in standard dimensions) are commonly used components in products, but products containing such components are not commonly perceived as modular products, although there in theory is little difference to LEGO products, which are commonly perceived as very modular.

Platforms and modularity is however all about how we perceive the products and the architectures and components, they consist of – especially if this perception decides how we manage them. Internally standardized components must be perceived and managed in the same way as screws and bolts.

Likewise the difference between common and individual modular architectures is only a matter of perception. Two different products sharing a single screw can be seen having different modular architectures, but they can also be seen as having the same architecture with two modules – one is the common screw and one is a generic place holder module containing everything else. The second perception has no point though; no company would build alternative product variants *based* on a screw.

In conclusion product families can be *based* on a common modular architecture or not (i.e. if not, they can be based on a building block platform or a cornerstone platform).

2.3. Levels of commonality

Different industries operate with very different levels of commonality, ranging all the way from completely different products to almost identical products. They all employ some degree of modularization, but the purpose of the modules differs depending on which end of the commonality scale (Figure 12) they operate near.

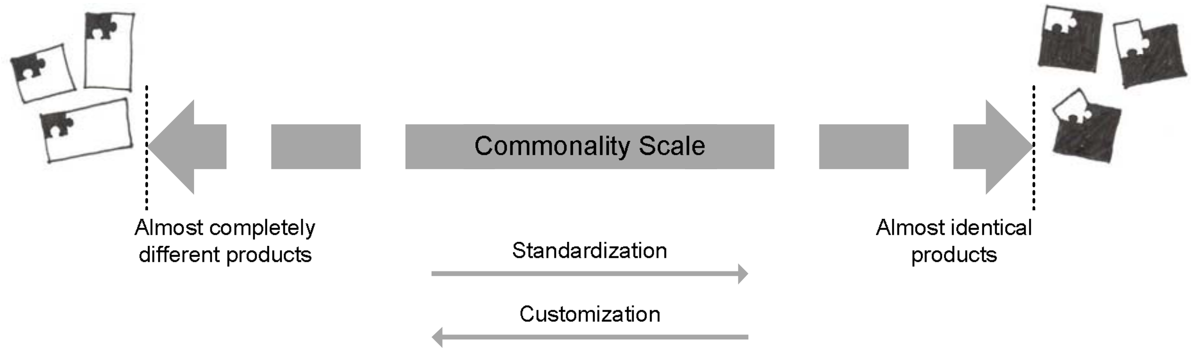


Figure 12: The commonality scale, which describes the level of commonality in a product family.

2.3.1. Standardization

Companies operating with a low level of commonality often have most to gain from platform-based product development. Standardization of components and processes and the creation of product families rather than separate product lines can possibly open up entirely new ways of operating for these companies.

When a company employs standardization, it typically identifies a number of components, which can be standardized into common modules for a number of products and/or projects (i.e. grouping of components which appear multiple times [Dahmus et al. 2001, Zamirowski et al. 1999]). The modules are in this case standardized units which are prepared for sharing (referred to as Standard Designs in [Harlou 2006]). In [Erixon 1998] this corresponds to the module drivers *common unit* and *carry-over*.

The modular architecture in this case is relatively simple. The modules confine that, which is to be shared. They connect to the remaining (otherwise integrated) product through simple standardized interfaces.

In [Robertson et al. 1998] it is stated that parts-standardization efforts across products may lead to the sharing of a modest set of components, but such a collection of shared components is generally not considered a product platform. And while this might be true, I will rather describe standardization as a simple first step in creating platforms in this thesis.

2.3.2. Customization

Companies operating with a high level of commonality will have a harder time increasing the level of commonality in their products – simply because many obvious *things*, which can be shared, are already being shared. These companies will typically want to customize their products.

It is however not certain that these companies have actually attained all the *effects* of commonality. There can be several reasons for this fact, but it often has to do with the fact that the commonality has not been introduced with the conscious purpose of gaining certain effects, and therefore the setup in the company's value chain has not been aligned with the customized products. I will return to this problem later.

When a company employs customization, it typically identifies a number of modules, which can confine the needed variety of a number of products and/or projects (i.e. grouping of components which are unique [Dahmus et al. 2001, Zamirowski et al. 1999]). The modules are in this case generic placeholders for a number of different units, which can replace each other. In [Erixon 1998] this corresponds to the module drivers *different specification*, *styling*, *technological evolution*, and *planned design changes*.

The modular architecture can be as simple as was the case with low commonality. The majority of the product or project is shared, and the modules confine the variety. The modules connect to the remaining product or project through simple standardized interfaces, which ensure the interchangeability.

2.3.3. Increasing the level of commonality

When describing the commonality of different products or projects with a low level of commonality, it is often easier to describe them by their similarities (i.e. the modules they share). Products and projects with a high level of commonality however are better described by their differences (i.e. the units they don't share – contained in the placeholder modules).

In many cases the modular architecture is however not as simple as this. Different overlapping patterns of commonality and interchangeability between some but not all products make descriptions of the architecture more complex. [Harlou 2006] provides an architecture diagram (called generic organ diagram), which help describe complex patterns of commonality, but does not go into interchangeability much. These diagrams are therefore especially useful in the lower end of the commonality scale, when a company is standardizing.

A possible next step in the process of increasing the commonality in a product family is the introduction of a common modular architecture describing the entire generic product (as opposed to only describing the shared modules in [Harlou 2006]) [Ulrich 1995]. A number of different tools can be applied for this. The *product family master plan* (PFMP) tool [Harlou 2006, Kvist 2009, Hvam et al. 2008] describes the architecture from a structural point-of-view resembling typical PDM structures – sometimes referred to as a generic BOM [van Veen 1990, Claesson 2006]. The *product family diagrams* [Van Wie et al. 2003] describe the architecture using 2D assembly drawings of a generic product. And finally the commonly used block-diagram approach, which describes functional elements or components from a flow/function perspective [Dahmus et al. 2001, Zamirowski et al. 1999] or interface perspective [Ulrich et al. 2000, Sahin et al. 2006]. These tools divide the entire generic product into a number of generic placeholder modules (some versions combine standardized modules and generic modules), which makes these tools applicable for very customized products as well, where having separate architecture diagrams for each product variant would be irrelevant.

2.4. Balancing commonality and variety

For the companies the primary purpose of offering different products is to meet heterogeneous user preferences [Ulrich 2007]. Customers are willing to pay more for the product that meets their needs best. Each product variant in a product family must therefore offer its own unique components or combination of components, which makes it special, and which matches a corresponding user or user group.

As a consequence it should always be considered, if introducing more commonality will remove the variety of the product variants. In many industrial cases however, there is a wide margin between the actual commonality and the potential commonality, which can be attained without limiting the essential variety between product variants.

2.4.1. Non-value-added variety

This wide margin of potential commonality is filled with *non-value-added variety*. Non-value-added variety is the kind of variety, which the majority of customers does not recognize, does not care about, or is not willing to pay the cost of. This variety does not drive product differentiation and variegation [Ramdas 2003]. It is essentially waste and should be replaced by common modules. Elimination of wasteful variety in products is in many aspects similar to the elimination of waste in manufacturing activities presented in lean manufacturing [Fiore 2005].

Non-value-added variety originates from a number of reasons – all of which has to do with missing or limited coordination of product development tasks or failure to update obsolete products. The typical reason is that different developers have developed the individual products and have not shared anything either because of limited or no knowledge of the other products or because they think it easier or better. Many large companies (even Toyota and Honda) experience parts proliferation and therefore spiraling costs due to these reasons [Desai et al. 2001].

Value-added variety on the other hand is based on one or more of the earlier mentioned module drivers *different specification*, *styling*, *technological evolution*, and *planned design changes*. It should however always be considered how many parameters based on true value-added variety a customer perceives. Big differences in one or more parameters will likely overshadow smaller differences in other parameters making them useless in effect.

2.4.2. Platform-based vs. innovative product development

Innovative product development and platform-based product development are from one point of view two different ends of a scale. Innovative product development is about introducing radically new products whereas platform-based product development is about introducing commonality between products (and thus limiting the amount of newness in a product).

These two approaches to product development are however not necessarily conflicting. They are more often merely different tools, which serve different purposes. Innovative product development is employed best for introducing new technology or new product concepts, which are unproven and risky. Whereas platform-based product development is employed best for reducing costs and order lead-time and increasing variety for already proved technologies and concepts. The two approaches should therefore be used in concert by companies – following up each of their innovative products with platform-based product families.

2.5. Generational commonality

In the previous sections commonality and variety has primarily been discussed from a snapshot of the product assortment at a fixed point in time. Sharing components involves however both choosing to design a component for use in multiple products and reusing a component in later products [Ramdas et al. 2003]. By describing variety and commonality from a time perspective as well as a market segment perspective we get a more real picture. In the time perspective we can have several generations of a product (generational variants or versions) and in the market segment perspective we can have several spatial (from market space) variants [Martin 1999, Martin et al. 2002].

The module drivers [Erixon 1998] mentioned in the previous sections can easily be combined with the generational and spatial perspective [Martin et al. 2002] (Figure 13).

The challenges of creating and adhering to modules caused by generational drivers are very different from the challenges of spatial drivers. With these, most platform initiatives are contained within one product development unit, creating standardized modules or generic placeholder modules for a family of products launched at roughly the same time can easily be decided upon and implemented.

	Generational perspective	Spatial perspective
Standardized module	Carry-over	Common unit
Generic placeholder module	Technological evolution, Planned design changes	Different specification, Styling

Figure 13: Combining module drivers [Erixon 1998] with generational/spatial perspective [Martin et al. 2002] and the standardized and generic placeholder modules.

Creating and adhering to standardized modules or generic placeholder modules for products launched in different generations can however be much harder. The primary reasons for this are that the requirements of future products can be hard to predict accurately enough and that decisions of this character typically rest upon specific persons, which may not be there in the future.

2.5.1. Versions and variants

Generally speaking different products belonging to a product family can co-exist only when a trade-off exist between them (for more detailed motives for variety see [Ulrich 2007]). If there is no trade-off, then one product is better than the other, and that other product is then superfluous and should be removed.

A trade-off however, can be based on many things. Sometimes technically identical products are offered under different brand names. The trade-off here is simply based on the brand names – and as long as some customers will prefer either brand name, none of the products are superfluous.

Products are also often developed in different versions over time but intended for the same marked segment and customer. In this case the new *version* of the product makes the older version obsolete and the older version should be removed from the product assortment.

In some cases the old version cannot be removed, because it is still in demand. This may be because of compatibility with other products, because of a difference in price, because of limited availability of the newer product, or other reasons. Common for the reasons is however that they actually reflect a trade-off for the customer, which actually makes them *variants* (though I might sometimes call them versions anyway).

Please note that while product variants may not from a theoretical point of view be superfluous, they might very well be so in reality. Quite often the costs of having an additional product variant outweigh the benefits of having this additional product variant.

2.6. Platform effects

As written in the beginning of this section the typical effects attained through platform-based product development are reductions in order lead-time and reductions of product costs. The promise of attaining these effects is often the beginning of a platform project.

A number of other effects have been reported in the literature. The primary tangible effects are dealt with in this section, while other vaguer effects such as minimizing complexity and product confusion [Moore et al. 1999] and product change [Ulrich et al. 1991] and improvement of knowledge management [Sanchez et al. 2001] will only be mentioned occasionally in the case study descriptions.

To describe the effects and how they relate to platform-based product development a reference model (based on [Blessing et al. 2002]) is shown in Figure 14.

As can be seen in the figure, some of the effects are attained in product development itself (those related to the product development resource load), while others are attained in other stages of the company's value chain (those related to downstream rationalization potential).

Platform-based product development ideally suggests that downstream rationalization is done upfront. As new products are planned, all required modules and components are analyzed and standardized, which hopefully leads to a reduction in the product development load (which again can be leveraged in a multitude of ways).

It is however not always possible to execute perfect planning of a future product portfolio upfront and a reasonable amount of after-rationalization must be planned for. This cleanup or "spring cleaning" in the product components and modules will not generate beneficial effects in product development (it may even require some extra work), but it may generate beneficial effects in the other stages of the company's value chain [Fisher et al. 1999].

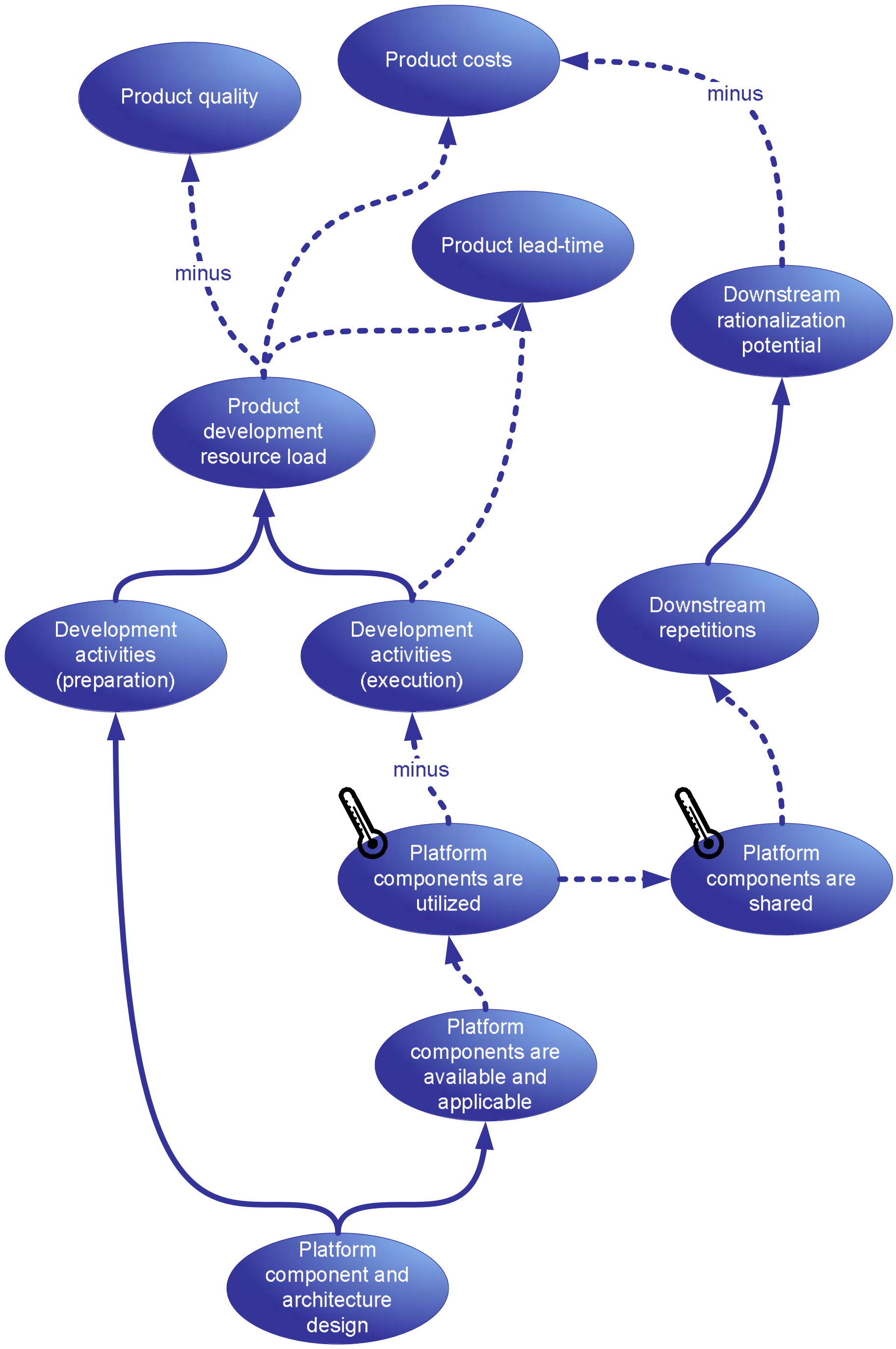


Figure 14: My reference model (based on [Blessing et al. 2002]) for the platform paradigm showing the intended effects of platform-based product development. Full lines are automatic consequences or effects, dotted lines are potential consequences, which must be enforced by management to appear. This figure will be explained in detail in the followings sections.

In general, the effects caused by products in other stages of a company's value chain (sometimes called dispositions) have been described in [Olesen 1992], but the focus in [Olesen 1992] is primarily about preventing unwanted effects. The deliberate effort of attaining specific effects in the later stages of a company's value chain has instead been described in [Andreasen et al. 2001, Andreasen et al. 2004] as an alignment of various architectures (e.g. the architectures of the products, the architectures of the manufacturing equipment, the architectures of the manufacturing processes, and the architectures of the logistical processes).

How to actually attain the alignment has not been described, but [Andreasen et al. 2004] states that architectures must be aligned on several levels of detail and that this is done through concurrent engineering of products, manufacturing process, etc.

Another aspect of alignment or lack of alignment is when different perceptions of the platform exist. When for example one department behaves as if the products of a product family are based on a specific platform, which another department does not recognize, there is a high probability of a mismatch in platform expectations and effects. The different perceptions of product platforms presented in section 2.1 through 2.5 can be used to align the strategies of different employees and departments.

Case studies described in [Harlou 2006] and the present thesis indicate that it is possible to attain specific effects through a deliberate effort of designing families of products with this aim (similar to DFX). In this thesis however, alignment and disposition-thinking will be regarded as a side-effect of introducing platform-based product development, where the main-effects relate more directly to sharing.

The downstream rationalization potential is therefore only a consequence of downstream repetitions in Figure 14, and downstream rationalization primarily leads to product cost reductions – even though it may very well be possible to achieve shorter order lead-time and better product quality as well as other effects in downstream activities based on the introduction of platform-based product development.

2.6.1. Cost reductions

Low volumes and small batch-sizes are costly and so is an ever-increasing amount of resources spent on product development and maintenance. The promise of reducing the number of unique components while increasing the amount of produced products is unparalleled for most product development managers.

Calculating the promised saving beforehand is however proving a difficult job. Although several calculation methods like total manufacturing costs, full manufacturing costs, and activity-based costs [Kaplan et al. 1998, Miller 1996] has greatly improved the way we can monitor product costs, it is still exceedingly difficult to predict the economic consequences of changes in the product architecture [Thyssen et al. 2006].

The reason is of course that assumptions of future product sales and supplier rates are extremely uncertain, and that few platform-based product development projects end up with a situation where the new products are directly comparable to the old ones. Additionally when products start sharing components it is not a straightforward task to calculate the costs of the individual products. Every cost cannot be meaningfully distributed to the products – especially when this includes several generations of products.

What remains, is the fact that developing common modules is an investment. This investment is likely going to pay off in the future – if the modules actually get used. If one module is successfully being developed and used in a number of products, it is clear that we are going to attain a number of synergies including saving development costs and investments and increasing our economies of scale [Ramdas 2003].

This is represented in the following way in the reference model. Economies of scale and similar downstream effects are labeled as downstream rationalization potential and the savings in development costs is a consequence of reducing the product development resource load (see Figure 15).

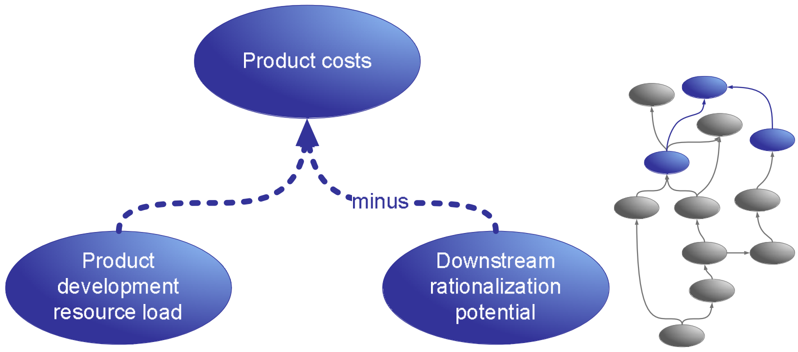


Figure 15: Subsection of the reference model Figure 14. Product cost reductions are based on reductions of the product development resource load and/or downstream rationalizations (incl. economies of scale).

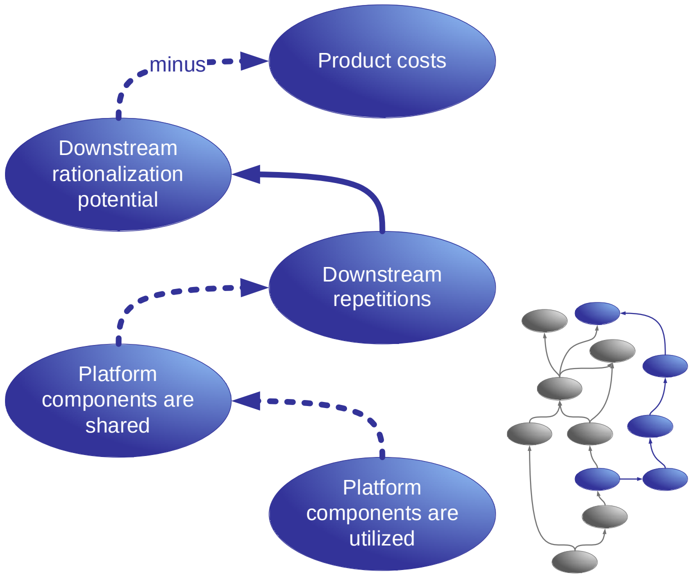


Figure 16: Subsection of the reference model Figure 14. Product cost reductions based on downstream rationalizations (incl. better economies of scale) is a consequence of downstream repetitions, which again is based on sharing.

Downstream rationalization can lead to reduced product costs if it is actively pursued. The obvious way of creating a downstream rationalization potential is by introducing downstream repetitions. Sharing platform components will often lead to downstream repetition by itself (i.e. reducing the internal variety will lead to higher batch-sizes), it is however important to check if this holds true in the individual company, since component sharing can sometimes be virtually invisible downstream because of lack of overview or poor documentation.

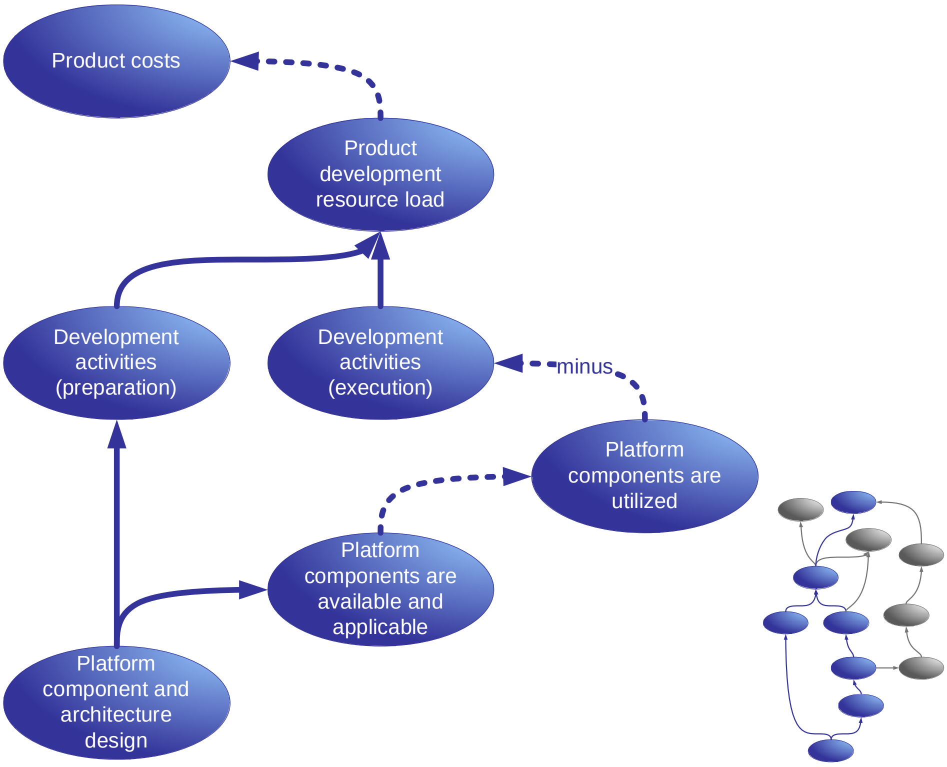


Figure 17: Subsection of the reference model Figure 14. Product cost reductions based on a reduced product development resource load is a consequence of reductions in the product development activities outweighing the increase in the platform development activities.

Reduced product cost can also be achieved through reducing the product development costs in companies where the development investments make up a significant part of the product costs (see Figure 17). Reducing the product development load can lead to lower product costs in these companies if management actively pursues it (i.e. it does not happen automatically).

Reducing the development load is possible by reducing the product development activities and this is possible by utilizing available platform components. Of course these platform components have to be developed – and the key challenge here is to utilize the platform components so much that the platform development costs are outweighed by the reduced product development costs.

One documented pit-fall of platform-based product development is that the standardization of components from several products into one common module can be too radical, so that lower-end products will get a cost increase from a too-high quality module, or so that higher-end products will be quality-wise poorer because of a too-low quality module [Meyer et al. 1997, Desai et al. 2001, Ulrich 2007].

It is especially the situation where low-end products get a cost increase which is dangerous because of the commonly occurring quality improvements which goes along with platform development (see section 2.6.4).

2.6.2. Time savings

The other major effect of introducing platform-based product development is the reduction in order lead-time. There are two components in this time reduction: One is an actual reduction of needed time to develop a product, and the other is an effect of dividing product development into a preparation phase and an execution phase (see Figure 18).

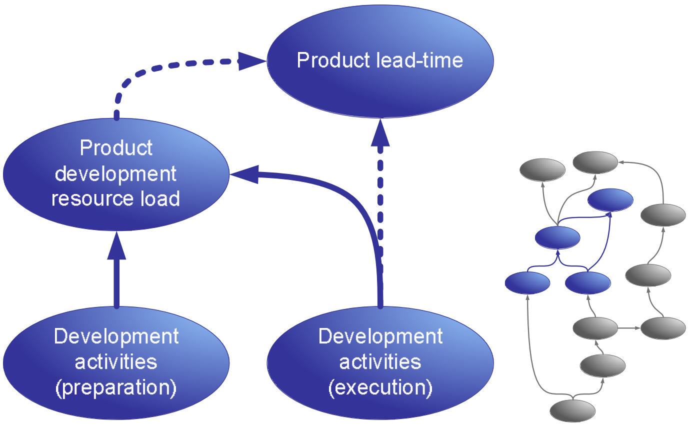


Figure 18: Subsection of the reference model Figure 14. Product lead-time reduction can primarily be based on reductions of the total product development resource load (true savings) and directly on product development activities (decoupling of platform development).

The real time savings relate very much to what I described with cost reductions in the previous section. Companies invest time in the development of modules – and are going to save time later, if the modules actually get used. Some companies fail to obtain this benefit because the same time is allocated or accepted for the development of the products, and product developers are prone to plan their time according to old time tables, even as they have fewer tasks. Management should therefore pair the introduction of platform-based product development with expectations of shorter product development time.

Figure 17 can be used again for reference simply by exchanging the top-bubble *product costs* with the other top-bubble *lead-time* from Figure 14.

The additional order lead-time reduction from dividing product development into a preparation phase and an execution phase is something completely different. These order lead-time reductions instead relate to the principles of lean manufacturing.

Companies, which operate in a market where order lead-time is a primary competitive factor, strive to diminish their order lead-time. Historically this has been done by building stocks of ready products. Products are produced for stock based on forecasts and customer orders are pulled from the stock. The *decoupling point* (i.e. the point from which orders are pulled) is pushed forward in the value-chain in order to decrease the order lead-time (Figure 19).

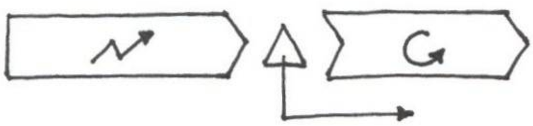


Figure 19: Traditionally decrease of order lead-time has been done by moving the decoupling point forward in the value-chain.

As the product variety and technological evolution has increased, this setup has become increasingly expensive. Lean manufacturing addresses this problem by abolishing all stocks and instead focusing intensively on streamlining the manufacturing process. In lean manufacturing the *decoupling point* is generally moved backwards in the value-chain, so that products are produced (or even engineered) to order and ideally pulled all the way through the value-chain (Figure 20).

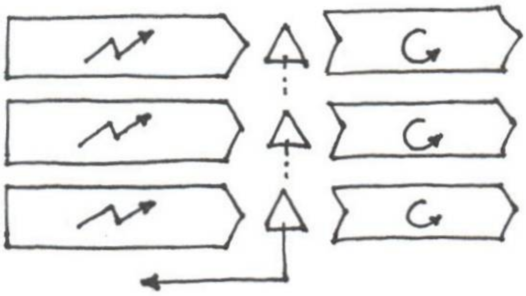


Figure 20: In Lean manufacturing the decoupling point is moved backward to minimize risks. The manufacturing process is instead streamlined in order to decrease the order lead-time.

From a platform-based point of view these two approaches can be combined. By creating a common stable platform we return to a situation where some degree of stock-piling is beneficial. The good platform is by definition less variable and less affected by technological evolution and therefore we can produce platform elements for stock. The actual products derived from the platform should however follow the lean principles, and this means that the decoupling point is moved backwards until it meets the *point of variegation* (i.e. the point where generic products turn into specific end products [Ramdas 2003]). This concept can be extended into the product developing phase, where the platform elements are developed beforehand based on expectations, whereas the unique components are developed based on orders.

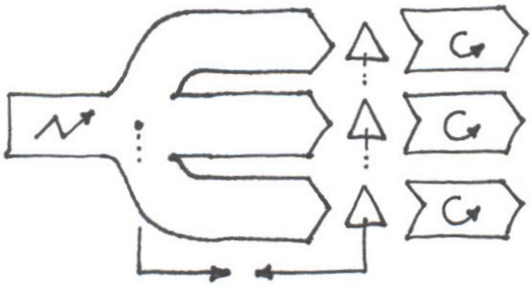


Figure 21: In the platform paradigm the decoupling point is moved backward to minimize risks and the point of variegation is moved forward to reduce order lead-time.

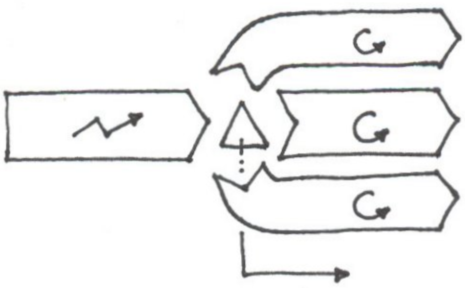


Figure 22: The platform paradigm manufacturing concept. The left arrow symbolizes the platform development phase, which is done beforehand based on forecasts and expectations (preparation phase). The right arrows symbolize the product development phase, which is done based on orders (execution phase). To reduce order lead-time the decoupling point (which is also the point of variegation) is moved forward.

As a common denominator we label all activities related to developing and producing platform components based on expectations as the *preparation phase* of the manufacturing process. All development and production of unique components based on orders are labeled as the *execution phase* accordingly.

It is only the execution phase, which makes up the order lead-time. The time saved here is however not an actual reduction of resources, it is instead a strategic move of the resources into the preparation phase, which is not directly coupled with the all-important product lead-time.

Reducing the order lead-time in platform-based product development is possible by moving activities from the execution phase into the preparation phase (i.e. moving the decoupling point forward), but this is only rational if we can move the point of variegation forward. This approach is sometimes referred to as late differentiation or the mushroom approach [Ulrich et al. 1991].

A strong separation of the preparation phase and the execution phase makes these two work processes very different. Ideally all technical problems can be solved in the preparation phase, while all marketing issues can be solved in the execution phase [Sanderson et al. 1995].

2.6.3. Increased product assortment

A number of platform literature authors also argue that a benefit of applying platforms in product development is the ability to launch more products. In my perception however, this is but one of a number of options resulting from lower costs and faster development time, which a company can decide on. Alternatively the whole platform paradigm can be regarded as having the purpose of introducing more product variants and suppressing the traditional side-effects (e.g. increased costs) of this [Hildre 1996, Berglund et al. 2005].

Companies such as Canon or Sony are often-used examples of companies launching a great number of product variants based on platform. But these companies could simply not launch this many products, if not product variant development costs and time had not been reduced significantly. Other companies may choose to spend the savings in costs and time elsewhere or simply realize the savings.

2.6.4. Quality improvements

Modularity leads to decoupling of development tasks, which will effectively decrease the complexity of the design tasks [Ulrich et al. 1991]. Also, sharing modules not only means saving resources. When fewer modules are developed we can (and should) use the more experienced product developers for these tasks. Combine this with a strategy of only sharing what is tested and tried, and we have a setup, where the quality of the common modules is very high.

The reason for testing and trying out common modules more than other components can be found in the following comparison:

- A company which is not creating common modules, but develops new components all the time, will have some components which fail and some which don't. In the larger perspective it is an acceptable cost to have a number of components which need to be redesigned and improved.
- A company which is using common modules will have relatively fewer units, which will fail or not. If these modules fail the cost impact will be significant compared to a situation where they don't fail. If the company is not willing to take this risk, it must

lower the probability of getting a faulty design by testing and trying the modules (more on this later).

Sometimes when common components are used in projects and time should be saved because of this, what actually happens is that the time remains the same but the quality increases. This is because product developers will not spent less time on projects if no less time is allocated to the project (see Figure 23).

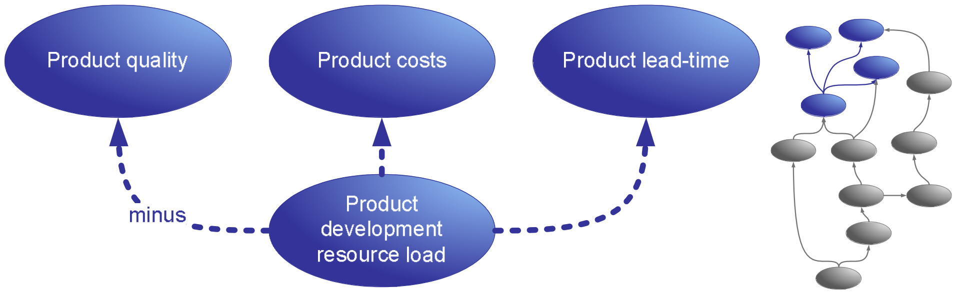


Figure 23: Subsection of the reference model Figure 14. Reductions in the development resource load can lead to several effects. If no less time is allocated to product development projects, but the activities are getting fewer (since platform components are utilized) then the increase available time for the remaining activities should lead to an increase in product quality.

2.6.5. Realizing effects

As stated several times in the previous subsections, the effects of platform-based product development depends very much on the role management is taking in the introduction of platform-based product development.

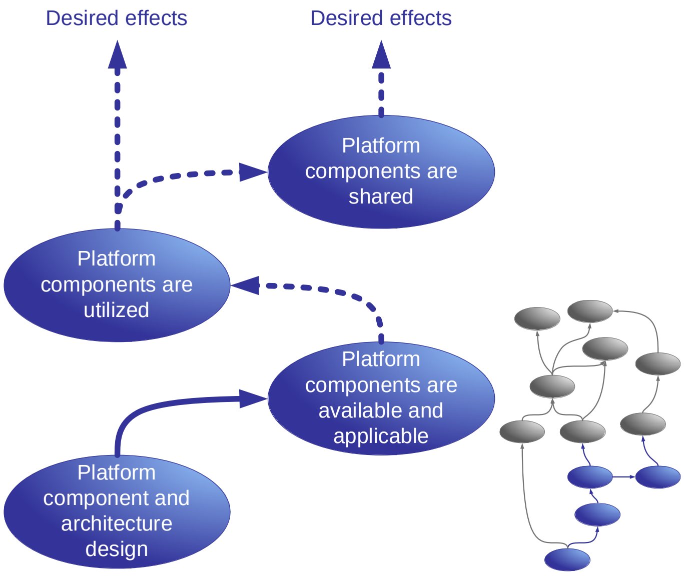


Figure 24: Subsection of the reference model Figure 14. All of the desired effects of platform-based product development rely on platform components being utilized (and some rely on them being shared in greater number).

Reductions in development resource load and rationalization potential in downstream activities can lead to great effects, but whether these effects are reductions in product costs, reductions in product order lead-time or improvements in product quality depends on management. Effects may also be completely unrealized and therefore wasted if management does not pair platform introduction with new performance measures.

Secondly all of the effects rely on platform components being utilized (see Figure 24).

However, making platform components available and applicable does not necessarily mean they will be utilized. If performance measures are not updated to include use of platform components, then there is a good likelihood that platform component utilization will remain at a minimum.

The only guaranteed outcome of a platform project is the platform project costs, and product development management should be careful to make sure this will not remain the only outcome of platform initiatives. Unfortunately this challenge has not been described in the platform literature so far.

3. Research Questions and Hypotheses

After having described the platform paradigm, I will now formulate some research questions which aim to expand the paradigm while building on what has already been described.

As mentioned in the beginning of this thesis, the responsibility of platform developers and platform developing companies is not only the development of a single platform, but rather to continuously develop new platforms and improve on existing platforms – continuously improving the platform assortment.

The platform paradigm described in the previous section as well as the major part of the literature related to platform-based product development offers no guidelines on this – instead it focuses on what kinds of platforms can be found, how these are created, and what the benefits of introducing them are.

3.1. The need for continuous development of platforms

Why is there a need to continuously develop new platforms components and improve on existing ones? I find it is the natural consequence of the ever-changing product portfolio and ever-evolving technology. As new products emerge and mature and as new production and distribution technologies are implemented – platforms will become outdated and irrelevant (see Figure 25).

As argued in the platform paradigm description, platforms are fundamentally based on the products we want to derive from them and the capabilities and options that are available when they are created. This means that platforms are only ideal when they are created. New information will constantly make the platform foundation obsolete – unless it is dynamic (see Figure 26).

The concept of a dynamic platform is contradicting in itself, because the platform is meant to be stable (i.e. not dynamic). If the platform is completely dynamic, most of the benefits of having platforms in the first place are forfeited. Economies of scale rely on a static modular system – and so does resource savings in general. Companies where these factors are an important reason for introducing platforms should beware of making their platforms dynamic.

Other benefits remain however. The concept of splitting the development task into a preparation phase and an execution phase (and thereby getting a shorter time to market) remains. And since the strategic decisions on which modules should be developed and used for which projects also remains, we are still likely to get a much more coordinated approach with less wasteful development of modules, which are too much alike.

The concept of continuous platform development, however, does not mean that all platform components should be completely dynamic. I only suggest that platform development does not end when the first platform-based product is launched. There will be a continuous need for planning and designing standardized common modules and design units which extends the platform or replaces elements of it.

From the impact model, Figure 26, it can be seen that the key challenges of continuous platform development will be about making sure that platform components are still shared (although to a lesser degree) and keeping the balance between the now increasing platform development activities and saved product development activities.

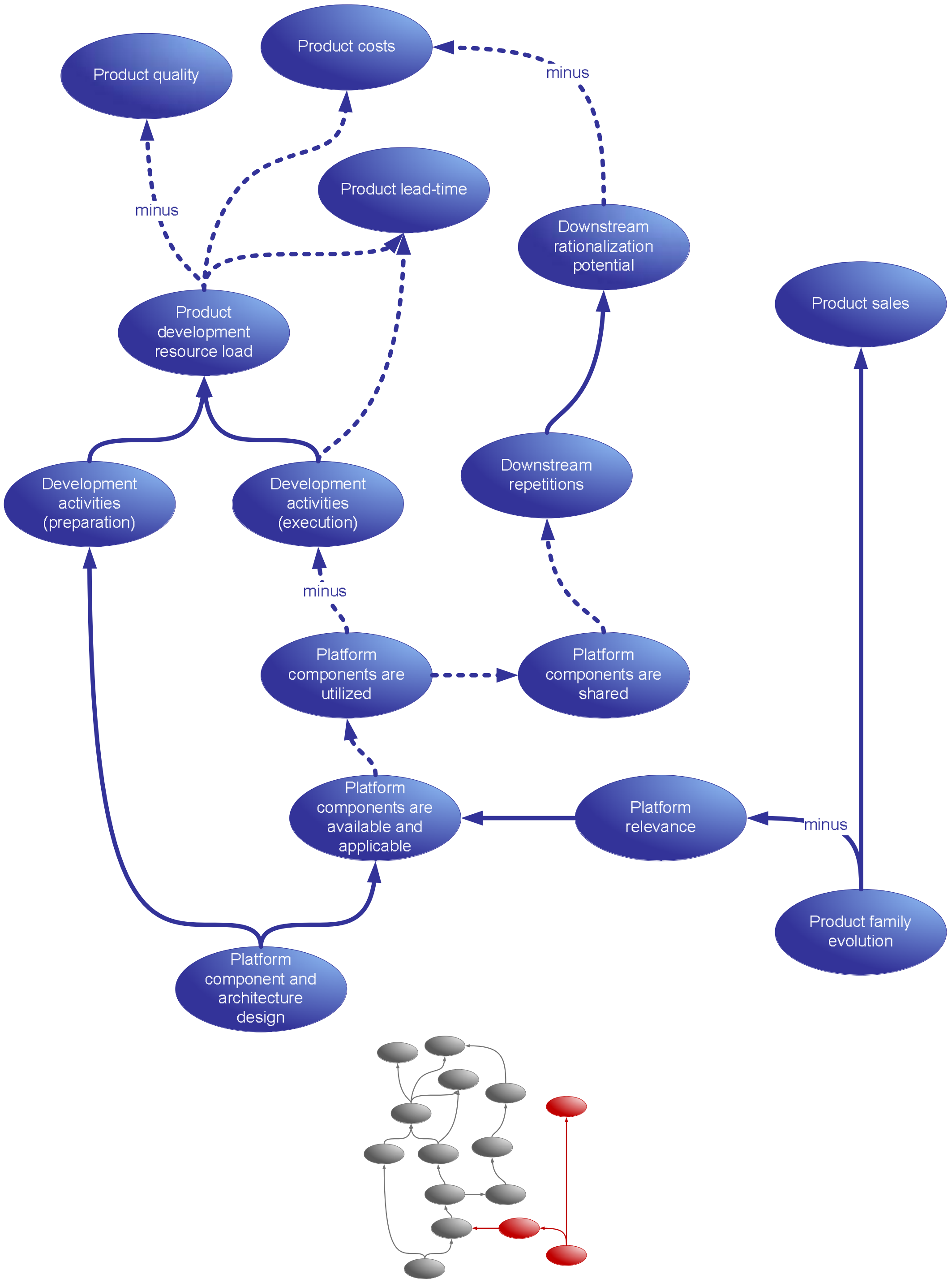


Figure 25: Extension of the reference model Figure 14. Product family evolution will lead to platform irrelevance, which will cause platform components to be inapplicable. If unchecked, natural product family evolution will with varying speed remove the foundation of the platform effects.

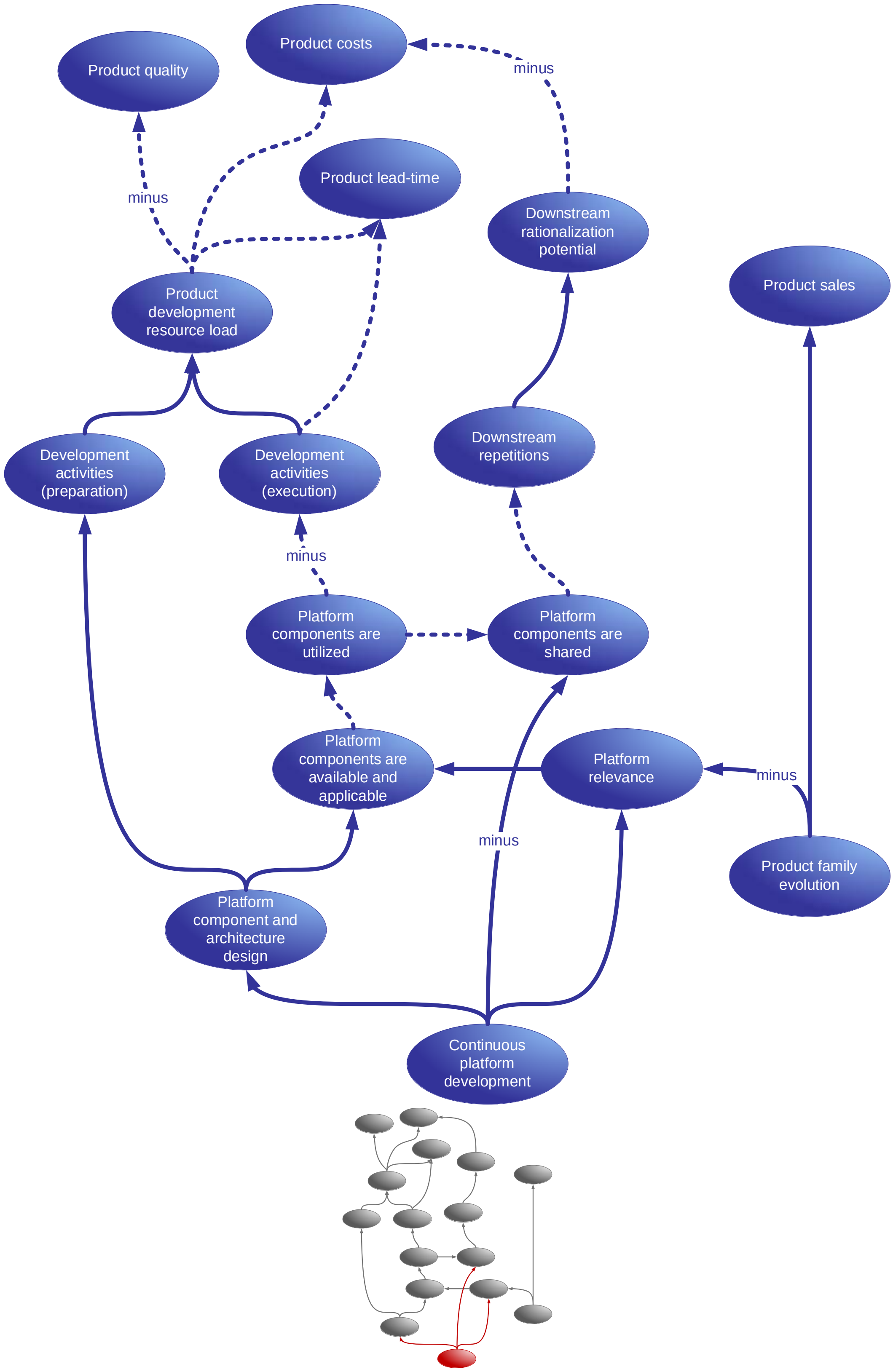


Figure 26: Impact model based on the reference model Figure 25. Continuous platform development will increase platform relevance it will however also increase the costs of platform development and reduce the sharing of platform components.

For those companies, which have less influence on what they will develop in the future (because it is dictated by their customers) – those that cannot decide – and for those companies, which value their innovative skills and need for flexibility – those that will not decide – semi-dynamic platforms is the only kind of platform, which will ever succeed.

Even for the other kind of company, which rely heavily on economies of scale and resource savings from sharing of all sorts of components, platforms will never be completely permanent. Eventually all platforms will become outdated – and new ones will have to be created – unless they are gradually improved.

Some companies – like LEGO (as I describe in this thesis) – cannot be put in one of these two categories. The LEGO element platform (i.e. the fundamental principles for LEGO bricks and how to put them together) is one of the most lasting platforms commonly known. The economies of scale and sharing of bricks between LEGO products are fundamental for the company's business and future survival. On the other hand some more specific groups of LEGO bricks – like the minifigure accessories (e.g. hats, tools and wigs for the LEGO minifigure) are ever-changing – and the company is unwilling to commit to pre-determined sharing of these components. Instead they have dynamic assortments of components, which still serve as platforms for the products, in the sense that they are ready and available for use in novelty projects.

3.2. Platform project ideas

With the need for continuous development of platforms comes a need for continuously finding and undertaking new projects for improvement of existing platforms and addition of new platform components.

The first research question of this thesis is based on this need:

How do we find platform project ideas?

To expand upon this question, I want to find out when and where platform projects start and who initiates them. What are the conditions platform projects are initiated under and what are the inputs to the process.

Because it is not possible to answer this type of research question completely, I will instead offer corresponding hypotheses.

In the platform paradigm section I described the platform as that which is shared between the products of product family. Platform projects must therefore be about finding new things, which can be shared between products, and since these things primarily are product components and architectures (see section 2) it would be natural if the product developers, who are designing these components and architectures, are also finding the new platforms.

Platform project ideas are found and suggested by the product developers themselves, when they are designing new products.

This hypothesis does however not suggest how the shared things that are not directly related to the products are then found. Nor does it suggest how different product developers coordinate sharing between products.

Assuming that many platform projects ideas with various levels of impact, effects, and consequences can be found in this way, and assuming that the company only has a limited amount of resources available for these projects, it will be necessary to decide between the suggested platforms.

How do we prioritize platform project ideas?

As stated in the platform paradigm section, it is extremely difficult to estimate the benefits of a platform project beforehand (see section 2.6). A commonly recognized analogy exists in the product development literature, where product concepts are prioritized by comparison (as in set-based product development [Morgan et al. 2006]), because it is not possible to objectively evaluate and quantify product concepts. I will assume likewise for platform projects.

Platform project ideas are prioritized by comparison with other platform project ideas.

These hypotheses suggest *an* answer to the corresponding research questions, which may not be the complete answer or the only answer. In my effort to confirm the hypotheses I will however hopefully find and describe possible approaches to finding and prioritizing platform ideas, which may serve as an inspiration to others, which is exactly the purpose of the prescriptive study.

3.3. Platform maintenance

Having responsibility of the platform assortment and continuous development of new platforms also means making sure the existing platforms perform as intended.

How do we monitor the performance of platforms?

State-of-the-art platform literature offers no guidance on how to do this. Since the purposes of platform projects are about gaining effects such as cost reductions and order lead-time reduction, it would however be natural to assume that these things could be measured and used to evaluate the platforms.

Comparing costs and order lead-times measured at different points in time is however extremely difficult. A very large number of different factors affect these values, and therefore it is near to impossible to conclude anything about a platform based on them. Instead it must suffice to measure appliance and sharing based on the models given in section 3.1.

Platform performance can be evaluated based on how and how much the platform elements are applied in the product development projects.

Since all of the beneficial effects are based on platform appliance, it logically follows that there is a positive effect on these factors if platform appliance is as intended (if performance measures are installed as advised in section 2.6.5).

Another aspect of platform maintenance is smaller additions to the platforms. In the platform literature it is normally assumed that new modules and components can be added to the platform later if they comply with the platform guidelines and rules. But what are the conditions for these small additions or revisions? Most additions and revisions – even small ones – will affect how other components of the platform are used.

How can additions and revisions be carried out without undermining the platform foundation?

Naturally this can be done by carrying out later platform projects that analyze and estimate the effects of such changes when they are needed. It would however be convenient if some underlying guidelines and rules could be put down, which would enable regular product developers to add and revise components while ensuring the platform foundation.

Specific rules and guidelines can be designed, which will allow additions and revisions to a platform, while ensuring the platform foundation.

Specific approaches in real cases will likely offer insights and inspirations.

3.4. Platform and product development synchronization

The literature describes platforms and how they are created. How they are actually used in the product development projects remains however mysterious. Sharable components are perhaps relatively straight forward – these should be designed into the products like any standardized component.

In reality this is not as easy. Assuming that the sharable components are essential parts of the products, they must be ready and tested before being used in product development projects – and remember the extra time developing shared modules of a high quality is likely to take. This does not correspond well with reducing product developing time and decreasing the response time of an order. This problem increases, when we are looking at dynamic platform development, where platform development is no longer a special event – but rather an everyday occurrence. It requires some way of synchronizing platform and product development.

How can platform and product development activities be synchronized?

First and foremost development activities should be split into preparation activities and execution activities as stated in section 2.6.2. Preparation activities are platform development activities and the execution activities is what remains of the regular product development.

Logically, the preparation activities must come before the execution activities, and ideally, they should be timed to end exactly when the execution activities begin. But how can this be implemented? As a working hypothesis I will suggest the following:

When a company develops platforms continuously, the platform projects runs in parallel with the regular product development projects, adjusted so that any platform project is always slightly ahead of the product development projects, which builds upon that platform.

This however, requires a lot of coordination. Again, specific approaches in real cases will likely offer insights and inspirations.

4. Case Material

The following sections give a short description of the various platform projects, which have served as a basis for this thesis.

4.1. LEGO Company

LEGO Company is very experienced when it comes to platform-based product development. The LEGO brick (a platform component – if there ever was one) is more than 50 years old, but still the company has experienced many problems related to poor coordination of product portfolio and product structure.

For the last couple of years a number of successful initiatives have been undertaken to improve the overview and support the management of the LEGO element assortment. LEGO's involvement in the research presented in this thesis is but one of these initiatives.

This section of the thesis is dedicated to describing platform projects at LEGO – old and new, as well as the background and setting they have been introduced into.

4.1.1. LEGO bricks

When other companies create platforms they often aim at modularizing their products, *just like LEGO bricks!* From an outside perspective LEGO products are perfectly modularized. An infinite number of LEGO models can be built out of the familiar and easy-recognizable LEGO bricks (Figure 27). Interfaces are perfectly standardized so that every two LEGO bricks can be joined in a number of ways.

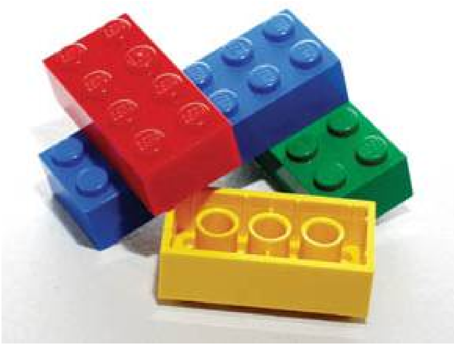


Figure 27: LEGO bricks, shared components that make up the core of most LEGO products.

Much of this is true. The interfaces are well-defined and classified (more on that later), and therefore LEGO bricks are not designated to each other, as product components of most other companies are. LEGO products are modularized, and in most LEGO products there are no or only a few unique bricks, even though the product itself can be entirely new to the customer. In this perspective LEGO has mastered perfect modularization, something most other companies can only dream of.

The oldest and most implemented platform elements in LEGO are internally referred to as the modular systems. In essence, the modular systems are sets of design rules which LEGO components have to comply with in order to be true LEGO bricks (i.e. they are therefore architectures). The modular systems can roughly be divided into three separate architectures: One for Technic bricks; One for Duplo bricks; And one for System bricks (i.e. the traditional LEGO brick), although some rules apply for all bricks. The poster describing the architectures, Figure 28, describes (from top to bottom) for each system the primary interface between components, the outer dimensions of the smallest unit, the alternative interfaces which relate to specific functions (e.g. hinges, bearings, grips), the size ranges, and the dimensional characteristic of the system (i.e. Duplo and System are stacking systems – bricks have a well-defined upside and downside, whereas Technic is a 3D system – no up and down).

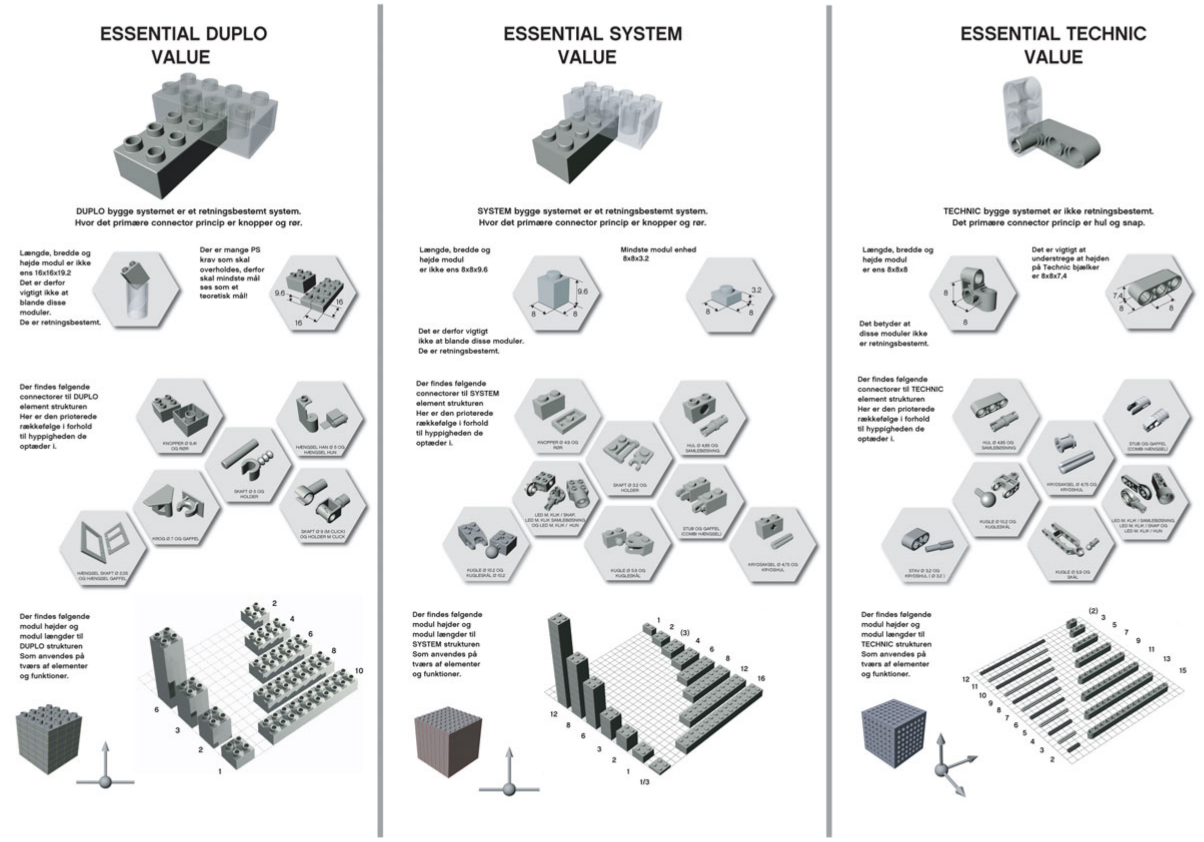


Figure 28: One of the most important overviews of the modular system (i.e. the brick architecture). The poster describes most of the rules that LEGO bricks must comply with.

The architectures do impose restrictions on the product designers, but these restrictions are based on modularity alone. They ensure that LEGO bricks can be connected and build together in a number of well-defined ways.

The original LEGO System architecture is more than 50 years old (although many of the alternative interfaces have been added later) and is well implemented in the organization. All but a few LEGO System bricks comply perfectly with the modular system, and these few exceptions are typically bricks with new special interfaces, special sizes or special purposes. The most noted exception is the LEGO minifigure (Figure 29), which does not comply very well with the System architecture.

The Duplo architecture was added in 1968 and resembles the System architecture, except it is double size. Duplo has fewer alternative connections, generally much fewer bricks, and more restrictions against small parts and sharp points, since the target user is younger. The Duplo architecture is not as well implemented since many Duplo bricks are not in compliance with the architecture. This fact can partly be explained by past organizational changes, but also by the fact that there is a balance between small children who actually favor a true modular building system and those who favor ready-made products.

Figure 29: Three variants of the LEGO minifigure. The LEGO minifigure is assembled by the customer from 4 different components (legs, body, head, and headdress) which is preassembled from 10 components (right leg, left leg, hips, 2 hands, right arm, left arm, main body, head, and headdress).

The Technic architecture was added in 1977, but has evolved significantly since its beginning. Originally it was simply an extension of the System architecture, and it is only recently that the final cut with the System architecture has been made. Today Technic bricks have a unique identity based on the Technic Snap and Snap-hole, without the familiar Studs and Tubes. The Bionicle product range (Figure 30) is part of the Technic architecture and complies with it, although many designers feel it deserve its separate architecture, and many customers remain unaware of this bond.

Figure 30: Bionicle products. Bionicle is a subsection of the Technic product range, although most customers probably remain unaware of this bond.

Other architectures have been designed through the years, many of them for products aimed at even younger children. Most of these have however only had brief existences and now only the three described brick architectures remain.

4.1.2. The burning platform

It is a well-established fact within change management that for an organization to change there needs to be a sense of urgency [Kotter 1996] sometimes referred to as a burning platform [Hiatt 2006] or a melting iceberg [Kotter 2006] – otherwise employees will be reluctant towards change and return to their old behavior when a project ends. At LEGO the burning platform which has enabled a change on a larger scale has to do with the modularization of the LEGO bricks.

The perfect modularization comes at a high cost. In reality there are far more LEGO bricks than most outsiders would know. It is easy to add new bricks to the assortment, which can be fitted on to all LEGO models. In an effort to continuously offer new products LEGO develops 200-300 new LEGO bricks every year (including color variants). This strategy is emphasized by the fact that it is generally the LEGO models with many new bricks, which are sold in high quantities, whereas the LEGO models with no new bricks are sold in significantly lower quantities.

Furthermore old bricks are not made obsolete, because they still fit into the new models, and who knows – we might need them in the future.

The costs of maintaining these bricks are not known explicitly, although some estimates have been given. Mistakes due to no overview, no sense of product, and the fact that item number is just a number has been estimated to 37,500 DKK for every new brick generated. Some claim that this estimate is much too low. Even more importantly an increasing number of LEGO brick molds are only used partially and put on stock for eternity. Quick estimates have shown that other molds will last more than 80 years with the current production rates.

The costs and complexity of the LEGO products reached a new climax in 2003. The economical results of the company in 2003, -888 million DKK, and 2004, -1,800 million DKK were disappointing – a direct consequence of realizing that all mold investments would not be paid (in 2004) and the staggering 12,005 active different LEGO bricks. The situation is more thoroughly described in [Clausen 2007]

The above mentioned complexity-costs and general escalation of complexity related costs and effects are similar to what other companies with complex product assortments are experiencing. But the perfect modularization of the LEGO products has not created the overview that would abolish the complexity. Instead it has encouraged a growth in the brick assortment which has only recently been stopped through a radical multi-facetted process spanning several years.

This cleanup and rationalization process is constituted by a number of different activities, where the most important ones are described in the following subsections.

Cleanup

The number of LEGO bricks was reduced from 12,005 to 7,416 by 2005. Most of these components had not been used in the previous 3 years, but since it was not unusual to sometimes reuse older components after a period of inactivity many product designer opposed this process. Other components were removed because of too high costs, too many problems in production, or too poor quality (e.g. complicated to build with, hard to distinguish from other bricks). Generally though, components were only removed as long as there were suitable substitutes.

Not only geometrical variants were reduced. Color variants were standardized and reduced significantly and different materials were also standardized and reduced in number. The number of external suppliers was reduced and production equipment was similarly reduced.

Exceptional reduction results were achieved, but the cleanup process left the component assortment decimated. Many gaps were now evident, while many poor components remained only because larger gaps would appear if they were removed. It was widely recognized by management, that similar painful cleanups would be needed in the future, unless more was done now. If product development behavior remained unchanged, then component growth would increase even more.

Component frames

The frames for developing new bricks in novelty projects are relatively new at LEGO. Their purpose is to limit the number of new components designed for new products every year, and thereby encourage reuse of existing components. Each component frame sets a limit to the number of new components, which can be introduced in a product development project.

The component frames are controlled by management and all projects must comply with these rules. Dispensations are sometimes given, but they are relatively rare.

Limiting the number of new bricks is important due to the earlier described uncontrolled growth problems, if we look at the individual bricks however, the element frames has another consequence. When limited to a small number of new bricks, product designers will often make these bricks truly unique to their product or theme in order to better distinguish their products and prevent cannibalization. This is a natural and beneficial reaction to limitations.

The bricks created in the novelty projects are as a consequence becoming increasingly specialized and are not very applicable in other LEGO products. Common bricks with a broad application are not being developed in the novelty projects anymore. This development does not seem to correspond very well to the *Shared Vision* strategy of 2006, where it is emphasized that LEGO should expand on the core. The *Shared Vision* also states that some of the key values of LEGO are the systematic creativity (i.e. being creative by using the system of modular building bricks in a unique way) and building experience, which can also be related to having more commonly applicable bricks than special bricks.

The component frames have helped cut down the uncontrolled growth, but when the bricks for development projects are being cut down, projects naturally think of them selves first and on the common good later. The component frames therefore cannot stand alone.

Component classification

It is essential for development projects to recognize the difference between bricks. Some bricks are special and exclusively used in one or two projects, while others are shared between many projects. And although all of the LEGO bricks, which comply with the brick architecture (described in section 4.1.1), are created so that they can be shared, it varies greatly how much they really are shared.

Bricks that are shared extensively are referred to as universal elements at LEGO. They typically resemble the traditional LEGO bricks and are very generic in appearance. The characteristics of the universal bricks are described in Figure 31.

Bricks that are not shared much are referred to as special elements at LEGO. They often have a very special appearance which relate to a special function or application. The characteristics of the special bricks are described in Figure 32.

Making a profitable LEGO product requires both universal and special bricks. But as shown in Figure 31 and Figure 32 universal and special bricks are two very different kinds of bricks. Creating a special brick and afterwards making it more universal will often result in an component, which is not great-looking according to the designers and will not be shared because it is still too specialized; it therefore holds no real value to the brick assortment.

Furthermore designers are primarily concerned with building products. Creating new bricks are only a means to obtain this primary concern. The creation of special bricks fits nicely into this setting. The purpose of special bricks is to improve *one* product. The creation of universal bricks however does not fit into this setting. Creating an element with a broad application requires a long term perspective and overview of all the novelty projects, something which most designers don’t need when designing their products.

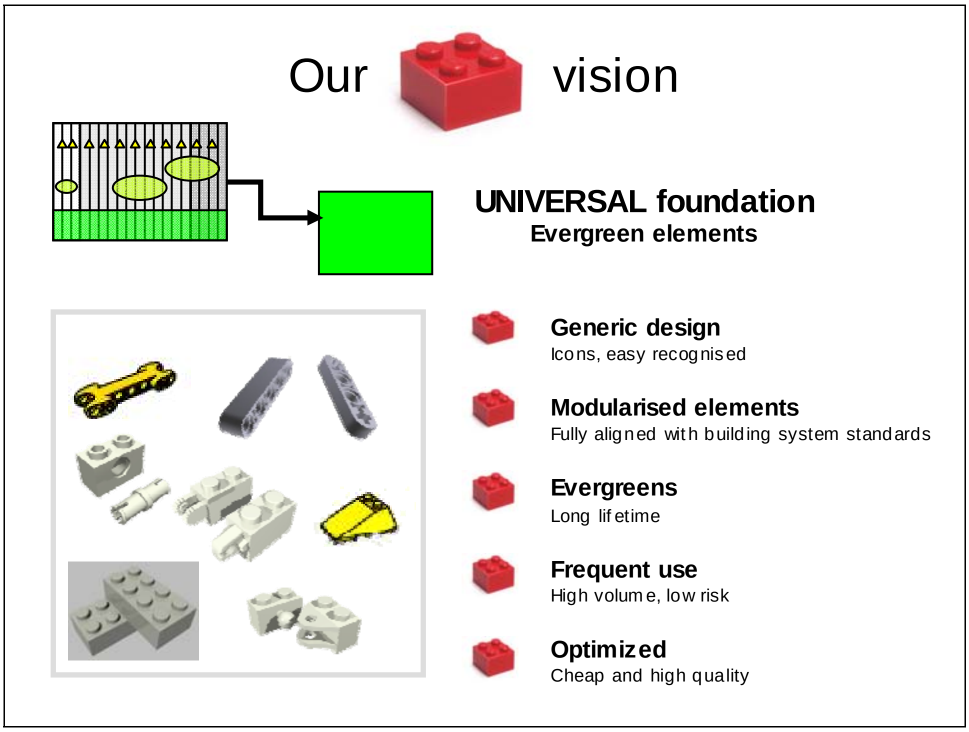


Figure 31: Slide describing universal bricks. Universal bricks (called universal elements internally) are shared between many products.

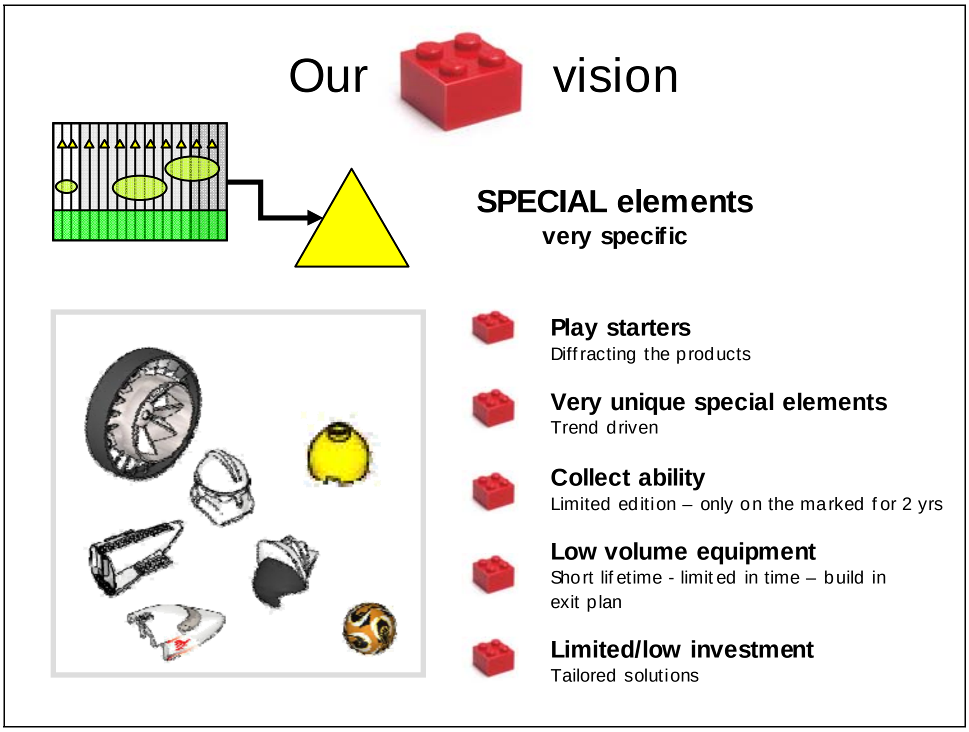


Figure 32: Slide describing special bricks. Special bricks (called special elements internally) are not shared between many products. They do use standard interfaces like studs and tubes, but often have a special appearance or limited interface possibilities, which in effect make them harder to apply.

Generic bricks or generic elements is the common classification for all bricks that too special to be universal, but still to universal to be special. Generic bricks are described in Figure 33. A major group of bricks are those that are theme-based, which means that they are only used in specific play-themes (i.e. a limited family of products). Such bricks are per definition not universal, but some of them can still be evergreen, and are therefore not special either – they are generic. Another kind of generic brick is a brick which can be used in all themes and many products, but is trend-based and only relevant for one or two seasons. This last group is however not very big and will not be dealt with in this thesis.

The generic bricks have lately been subjected to several coordinated revisions, where a group of bricks have been replaced or improved (these projects are described in the following sections). The reason that it is often generic bricks, which is subjected to these revisions, is typically that the universal bricks are performing well on their own, running in high volume, and the special bricks are too temporary to worry about revisions.

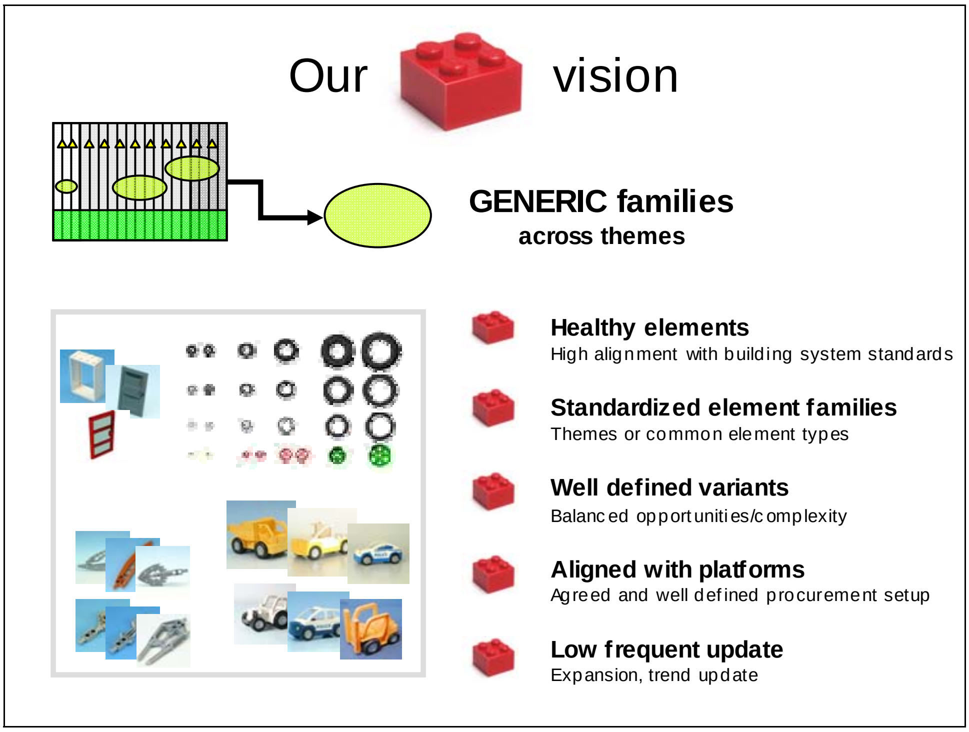


Figure 33: Slide describing generic bricks. Generic bricks (called generic elements internally) are too special to be considered universal and too universal to be considered special.

An overview of the correlation between sharing and the Universal-Generic-Special (UGS) classification is given in Figure 34.

The bricks are classified by a group of senior designers as universal, generic, or special when they are first designed. To assist this classification a review sheet has been designed. In short, the review sheets list a number of criteria for universal bricks (one is compliance with the corresponding brick architecture), and if the brick in question complies with almost all of these criteria it is classified universal, if it complies with many of the criteria it is classified generic, and if it complies with less criteria it is classified special. Bricks, which comply with no or only a few criteria, are generally advised to be redesigned.

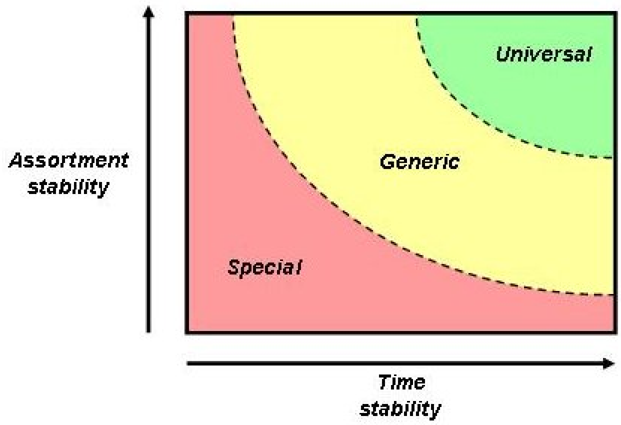


Figure 34: Table showing the correlation between sharing and LEGO classification.

By introducing the UGS-classification, LEGO is making the product designers recognize the difference between components. Special bricks are short-term investments that need to be repaid fast, whereas universal bricks are long-term investments. The company must act accordingly when a product designer chooses an existing component or designs a new component – making short-term investments in disposable equipment for special bricks, and long-term investments in reusable equipment for universal bricks.

The UGS-classification is based on component application. Components which can be shared in many products are universal. Application in many different products however, does not mean that the products are actually being produced in high volume, since various LEGO products are produced in very different volumes.

For cost and investment purposes, the UGS classification is therefore combined with a high/low volume classification and a color classification (the UGS classification is based on the geometrical design – not color).

Component costs

The product development projects are already responsible for the economical success of their products and therefore very concerned about the costs of the bricks they are using. Each project has a budget for ‘buying’ LEGO bricks for their models.

Internally, LEGO bricks prices are adjusted based on their classifications. Investments in molds are based on UGS classification alone, and production costs are adjusted based on a combination of the UGS, the color classification, and the high/low volume classification (see Figure 35).

The payment of mold investments is spread out on the produced bricks. For special bricks the payment per brick is calculated based on forecasted production quantities, whereas it is based on the technical performance of the mold for generic and universal bricks (i.e. it is assumed that the mold will be exhausted for generic and universal bricks).

Production costs are affected by the volume bricks are produced in. Small batches means more mold setup and cleaning, therefore bricks are given a between +20% and -15% adjustment on their cost per hour of molding based on their UGS-classification, color classification and high/low volume classification. Additionally and also based on these classifications, an obsolescence bonus of +5% is added to the material costs of the special bricks. This bonus is based on the fact that forecasting is more uncertain for smaller quantities, and LEGO is generally scrapping more of these elements due to excess production.

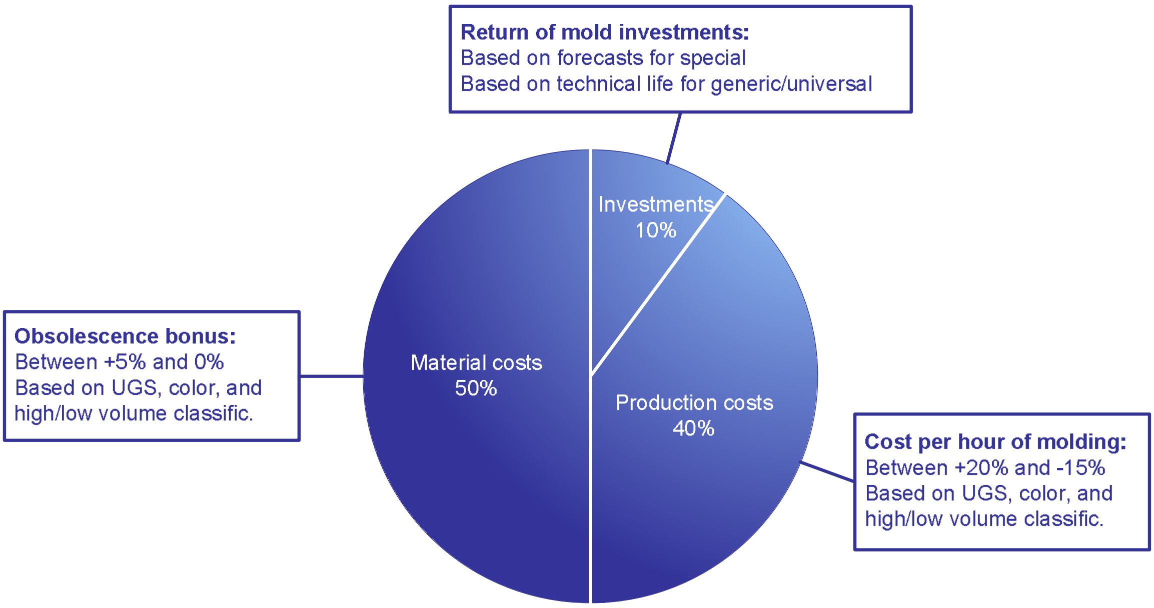


Figure 35: Cost division of a typical LEGO brick with adjustments based on internal brick classifications.

The costs of a typical LEGO brick are 50% materials, 40% production costs, and 10% investments as can be seen in Figure 35 – although this varies a lot from brick to brick. The UGS classification has a significant cost impact, but it is not overwhelming on its own.

The huge difference is the mold investments, which can be devastating for projects with low forecasts, where the investments make up a significantly larger proportion of the total costs. This way of returning the investments is however essential based on past experiences with unreturned investments.

In the past cost structure, where investments were always based on how many bricks a mold could produce in its lifetime, special bricks were a significant risk. LEGO risked not getting a return on their investments. Today when special bricks are paid for upfront, it is the universal and generic bricks that are risky (although to a lesser degree) – they must be reused if the investments are to be returned.

This has made the UGS classification much more important.

Aside from the costs based on classifications, there are several startup fees, which the projects must pay from their budget. Creating new bricks or using other colors or materials for existing bricks costs a small but not insignificant fee, which correspond to the complexity costs mentioned in the beginning of this section. Because of this it generally costs less to reuse available bricks than creating new ones.

Because the cost adjustments and fees relate directly to actual costs in manufacturing, the classifications give no true penalties or discounts, but only makes up for costs, which cannot be allocated to bricks in the traditional way. The past growth in special bricks has been partly due to unrealistic low costs, which has now been amended with the UGS classification. The projects still design primarily special bricks however (although much less) even though they must pay the adjustments and fees, and the number of universal bricks designed in product development projects remain few.

4.1.3. Platform projects

The strategic line at LEGO dictates, that the company must grow the core and emphasizes the need for and value of the core building system. Growing the core and expanding that, which is shared is a cross-organizational responsibility. The individual product development projects *cannot* take this responsibility, because they don't have the cross-organizational knowledge about needs and wishes in other projects, and they *will not* take this responsibility, because their objective is to make unique and selling products.

This means that there is a need for cross-organizational projects, which will take responsibility for creating and revising that, which is shared. Many such projects (most of those, which deal with the bricks and products themselves) are undertaken by the organizational unit DesignLab.

DesignLab is a group of senior designers and supporters, who have responsibility for maintaining the brick assortment. Traditionally this has been done by evaluating new bricks and advising other designers on brick development in design reviews as well as showing and documenting the brick assortment in the LEGO brick library, which is a wall, which shows all the LEGO bricks.

Since 2005 DesignLab has also undertaken a number of platform projects, which I have taken part in and use as the core part of the empirical studies for this thesis. Most of these projects are part of the list in Figure 36.

The platform projects listed in Figure 36 can be roughly divided into four groups. One group is the projects, which began before the first screening project in 2005. These projects vary greatly in scope, purposes, and the people involved – both from each other and from the later projects.

The second group is the screening projects. These three projects have formed a frame for the remaining the projects. The purpose of the screening projects was to find and evaluate platform project candidates. It is the thoughts and learnings from these three projects that are my primary inspiration and it is in the latter two, that I have participated the most. The screening projects are yearly processes and it is currently continuing in 2008, they give in my opinion an example of state-of-the-art continuous platform development.

The third group is constituted by various platform projects, which have been carried out as part of the annual screening process. Some of the projects extend over several years and can therefore be part of the project portfolio for several screening projects. These platform projects are not the classical platform creation projects, where a platform is introduced into an organization, which did not employ platforms beforehand. It is rather platform revision projects, which change the architecture and the shared components in order to improve the performance of an already existing platform.

The fourth group is constituted by the universal elements projects. These projects have also been carried out as part of the annual screening process as the third group, but where the projects in the third group focus on revising a specific part of the existing platform, the universal elements projects focus specifically on adding new common bricks in various parts of the brick assortment.

Projects	Time frame	Description	Researcher participation
Low Volume Project	2002-2003	Guidelines and revision of molding equipment	Interview
Module Mold Project	2004-2005	Guidelines and revision of molding equipment	Minor participation
Wheels Project	2005-2006	Cleanup, guidelines, and additions for subgroup of bricks	Full participation
Screening 2005 Project	2005	Finding and screening platform project ideas	Minor participation
Power Functions	2005-2007	Addition of new subgroup of bricks	Interview
Screening 2006 Project	2006	Finding and screening platform project ideas	Full participation
Minifigure Accessories Project	2006	Cleanup and classification for subgroup of bricks	Full participation
Ball Joint Project	2006	Revision and improvements for subgroup of bricks	Interview, Framing
Walls & Windows Project	2006-2007	Cleaup, revision, and additions for subgroup of bricks	Interview, Framing
Universal Elements Project 2006	2006	Addition of various universal bricks	Full participation
Screening 2007 Project	2007	Finding and screening platform project ideas	Full participation
Shapes Project	2007	Guidelines and classification for subgroup of bricks	Full participation
Frames Project	2007	Guidelines and additions for subgroup of bricks	Interview, Framing
Panel Project	2007	Revision, guidelines, and additions for subgroup of bricks	Interview, Framing
Universal Elements Project 2007	2007	Addition of various universal bricks	Full participation

Figure 36: LEGO Platform projects documented in this thesis to a smaller or larger extend. Most of these were undertaken by DesignLab between 2005 and 2007.

Platform projects before the first screening project

Initially formalized platform projects at LEGO were undertaken by the production and supply chain part of the organization. Their purpose was to create alignment between the products and product components on one side and the production and distribution systems on the other side.

The *Low volume mold (LVM)* project succeeded in creating a range of new mold concepts, which made it possible to design cheaper molds with a shorter life. LEGO bricks, which would never need to be produced in the volumes that traditional molds were capable of could now be produced much more efficiently (i.e. with far less investments). In this way the manufacturing capabilities had been aligned with the needs of product development.

The project was a success, which has led to number of consequences. First and foremost it improved and made possible a number of specialized products (e.g. Bionicle, Star Wars, and

Harry Potter product ranges), which would otherwise have suffered greatly under the costs of the many special bricks, which are in these products. Without the low volume molds these projects would probably have been terminated or out-sourced to China when the new cost structure was implemented, with dire consequences for LEGO Company.

Secondly the molds actually looked so inexpensive, that product designers started aiming for these molds in general. Part of the *LVM* project had been to design visual guides for the product designers, which would communicate the limitations and capabilities of the molds, and designers actually started modifying bricks, in order to make them fit the new mold concepts. Within two years more than 50% of the new molds created at LEGO were low volume molds.

Because of the success of the *LVM* project, it was decided that more molds should also be designed in a similar way. This was the purpose of the *Module mold* project.

Like the *LVM* project the *Module mold* project created a range of standardized mold concepts, but now with the aim of encompassing the entire spectrum of needed molds. The project succeeded so far, that today a large group of molds (although not all) – including *LVMs* – are *Module molds*. In mold performance however the *Module mold* project made only minor improvements and failed to revolutionize the mold assortment the way it had been done in the *LVM* project.

On reasonable conclusion from the *LVM* and the *Module mold* projects was that it was only possible to achieve relatively small improvements on the costs of the entire product assortment, when the products and product components themselves remained unchanged. The *LVM* project had influenced the product components to some extend, but the conclusion at the time was that it would be much more beneficial if platform projects involved product development directly. Involving product development would also mean that future forecasts and development pipeline data could be added to the data material, rather than suffice with past facts and tendencies.

Projects	Time frame	Sub-assortment	Architecture	Components	Results
Wheels Project	2005-2006	Approx. 100 components reduced to 50 components	Strict: All components are predefined	Best existing components survive, others are replaced, new ones are added	Reduced investments, reduced maintenance, improved quality

Figure 37: The Wheels project summary (for later referencing).

Late 2004 the *Wheels* project was started as a pilot project. The project was planned as a cross-organizational project with team members from product development and the supply chain. The purpose of the project was to reduce complexity. The assortment of LEGO tires, hubs and rims for LEGO products had slowly been increasing till 101 different components by 2004. Each year a few new components were added, but no old components were discarded. Complexity costs were not overwhelming, however as more and more components existed, other costs increased as well. More components meant more mold changes, more stocks and more forecast errors, but it also meant less utilization of older molds (unwanted effects are shown in Figure 38). In the *Wheels* team we found that the investments in wheel component molds each year were twice the amount actually required if all molds would be exhausted. Most molds would never be exhausted before their production quantities were gradually reduced to almost nothing.

More importantly, it was determined after a number of interviews with product developers, that wheels in general were in fact not selling products. This meant that the number of wheel component variants should be minimized and based on necessary requirements. Wheel components should be standardized so that they could be shared to a greater extend.

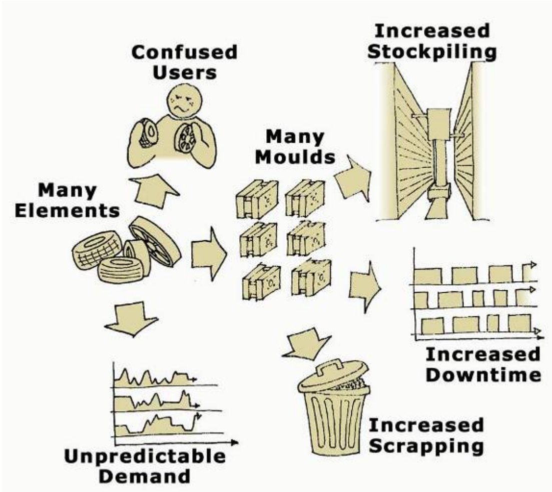


Figure 38: Primary unwanted effects (internally and externally) of having too many wheel components. These unwanted effects were perceived as a problem and the wheels platform project was initiated to try to eliminate them.

One additional challenge of the *Wheels* project was that similar projects for revising the wheel assortment had been initiated several times before. Each of these projects had failed however, primarily because of lack of commitment in the organization.

Initially, the team defined a framework for wheel components. A matrix of 46 components was created, some of the best old components were used, but also new components were created to fill out the 46 spaces. As can be seen in the right side of Figure 39, the new wheel components were initially only specified and designed. Some of the components were also created within the projects (those that were to replace existing components in use), but others were left for future product development projects to develop within their component frame. The primary difference between the new variants was sizes, since it was not considered possible to share the same size wheels between different sizes of vehicles.

Future bricks development could also replace some components rather than add to the assortment, which in itself would make the creation of new components much harder for the product development projects, because they would have to create a new component that could substitute the older component in all its applications.

The *Wheels* project limited the number of new future component and reduced the number of existing components to increase the utilization of molds. Many minor adjustments of designs were also implemented to improve the performance of the production equipment. In this way the product development and product assortment had been aligned with the manufacturing capabilities.

In parallel with these activities a number of project presentations were carried out. The purpose of these presentations was primarily to gain approval from management for making the initial investments in new molds. By continuously presenting and highlighting the benefits of the project for management, product developers, supply chain employees, and other stakeholders we succeeded however not only in gaining approval from management, but also general commitment from the entire organization. These activities related to the implementation and anchoring of the *Wheels* project has been documented in [Fiil-Nielsen et al. 2006].

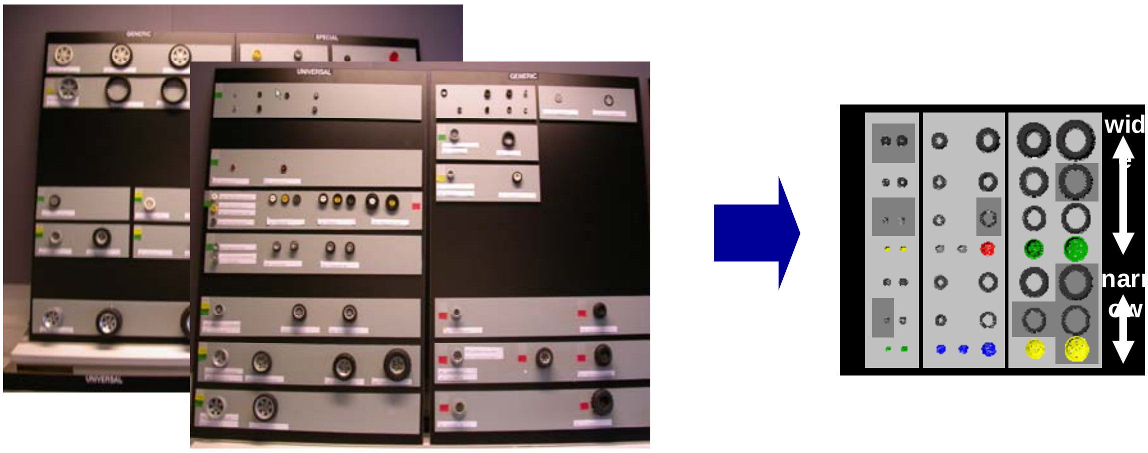


Figure 39: In the Wheels project the existing assortment of wheel components were revised and changed into a matrix of standardized components. Initially the new wheels were merely specified, but today most of the new wheels have been created and the old ones have been discontinued.

The primary result of the *Wheels* project has been a significant decline in investments, and the investments have fallen to a level just above what is required for replacing exhausted molds.

Wheel components are still being discussed, and this indicates the importance of the components. The project was not an easy success, but a rational compromise, which could only be attained by a cross-organizational project. The initial matrix has been changed after the project ended in 2006 and a few new components have been added to the matrix. In my mind this confirms the purpose of the project, since it was never intended to halt the development of new components, but merely ensure that new components were added with good reason.

The screening projects

The first successful platforms described in the above had been undertaken as independent projects. It became clear, though, in the second half of the *Wheels* project that the implementation of the project was very much dependant on the concurrent product development projects. In a busy development organization designing product components, which would not be used in the upcoming products, simply had too low priority. Consequently, the components, which had only been designed on a relatively low level of detail in the platform project, were not completed until they would be used in a product.

Also since LEGO was beginning to develop several platforms simultaneously, any funding of individual platform projects would ultimately be based on priority. This had led management at LEGO to demand one single process, in which all platforms could be presented at the same level, so that they could be compared and prioritized.

To make the timing of the platform projects even better it was decided to follow a more structured and manageable process, which could be synchronized with the main product development in such a way that the results of the platform projects would be available to the product development projects.

The overall process, which can be seen in Figure 40, follows the pre-project part of the current development model at LEGO, the *LEGO Development Plan* (LDP). The LDP is the product development process, which all product development projects (called novelty projects) at LEGO follow. The pre-project part is a three-milestone stage-gate process, where novelty projects are gradually detailed and evaluated.

Figure 40: The stage-gate process for the screening projects based on the pre-project part of the *LEGO Development Plan*.

At the P1 milestone the first preliminary projects are presented. At this stage only rough ideas and marketing aims are presented. At the P2 milestone more detailed projects are presented. Concept drawings and economical guesstimates are given. At the final P3 milestone even more detailed projects are presented. Prototype models are presented, and detailed consumer feedback is given.

After the P1-P3 milestones management commits to the remaining projects, and the actual product development starts. A new set of milestones frame this process, but projects are rarely rejected after the pre-project part of process.

The screening projects are also partly described in the earlier publication [Mortensen et al. 2009].

Screening 2005 project

From March until October 2005 LEGO made their first attempt to synchronize the development of product platforms with the regular early-stage novelty projects of 2007 products.

The project team responsible for the process was partly self-appointed platform stakeholders and key-employees from product development, from the supply chain organization, and from the traditional platform department.

The tasks of this project team were partly to generate the actual platform candidates and partly to create the process itself which would make this possible. Much emphasis was put into generating the process, since the general perception was that there were a multitude of platform opportunities.

A number of interviews in the various departments generated these platform candidates. In total 42 candidates were identified. These platform candidates included:

- Technological projects, which would introduce a new technology or function to the assortment in a number of new LEGO elements;
- Packaging projects, which aimed at replacing existing packaging material or equipment (i.e. Not product platforms – but packaging platforms);
- Clean-up projects, which aimed at restructuring and repairing specific segments of the element assortment which seemed to complex or big (similar to the Wheels platform project described previously);
- And material substitution or reducing projects and production alignment projects, which were to reduce the cost of specific costly segments of the assortment.

Additionally, several of the generated platforms were combinations of these four types of projects.

Based on previous experiences it was decided to synchronize all product platform development with the novelty projects. The platform candidates were taken through the same milestones as the novelty projects, but they were slightly more detailed and completed throughout the process, so that any result could be available when they were needed in the novelty projects.

For each platform candidate a rough two-page platform business case template was created (See Figure 41, left). This first business case included a concept description, a rough cost/benefit assessment, a list of potential users, as well as rough risk and feasibility indicators. At this point all of these figures were simple assessments.

In June, at the P1 project milestone, the gross list was presented for the product development management, who rejected all but five projects. The reasons for most of these rejections were primarily lack of alignment with company strategy and lack of business potential, but poorly communicated concepts may also have been the case. The high rejection rate increased focus on the remaining projects, but since these now had to move on to the economical assessment of the platforms and get ready for P2, the fundamental concept descriptions were never improved.

At P2 in September two more projects were closed. Only three projects remained, and two of these had poor perspectives. Most of the people who had originally participated in the platform projects were transferred to other tasks, and the platform process had lost all momentum. Finally, the central organizational unit behind the platform interest group was closed and the overall project discontinued.

Only one project survived the P3 milestone, mostly due to the fact that the project team had successfully made it independent from the platform interest group and had from the very beginning had a very strong concept.

The 2005 platform development process had failed. The theoretical approach to platform-based product development, which had resulted in a process based heavily on methodology and grand master plans (like in Figure 41), had failed. Future platform projects would again have to justify themselves and a more practical approach was needed.

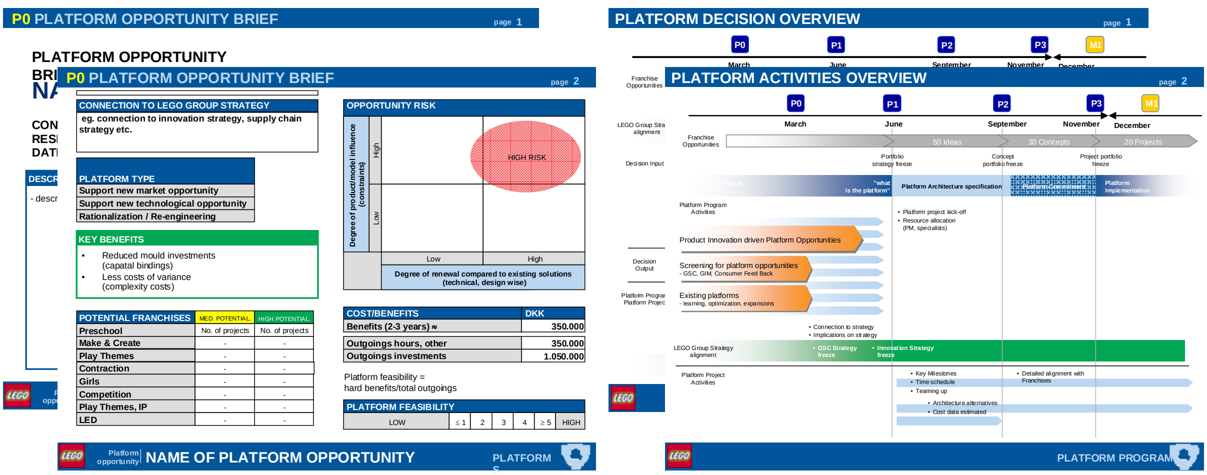


Figure 41: Left: The 2005 P0 Platform Opportunity Brief (P0 is no true milestone). Right: The master plan of the 2005 process, which described how platform candidates were found and how they should evolve between the milestones. Slides are shown for overview only.

Screening 2006 project

In the following year, from March until October 2006, this process was repeated after having been greatly refined.

The platform candidates in 2005 did lack alignment with the company strategy and designer requirements, and they did lack business potential. It remained, however, unclear, if some of the candidates could have achieved these things, if they had been given more attention.

The sheer amount of candidates had from the beginning of the 2005 process hampered the development of the individual projects. Likewise, management was quick to discard a candidate if it did not meet every criterion because of the multitude of platforms presented for them. Finally, the focus on the process itself took away the attention from the individual projects.

The individual candidates suffered from a lack of focus, and much time was spent going through old information and other repetitions because nobody could remember all the facts of the individual candidates.

To remedy this it was decided, even before the start of the 2006 process, to limit the number of platform candidates significantly. This would hopefully increase the quality of the candidates, and with a limited number of candidates focus could be retained throughout the whole process.

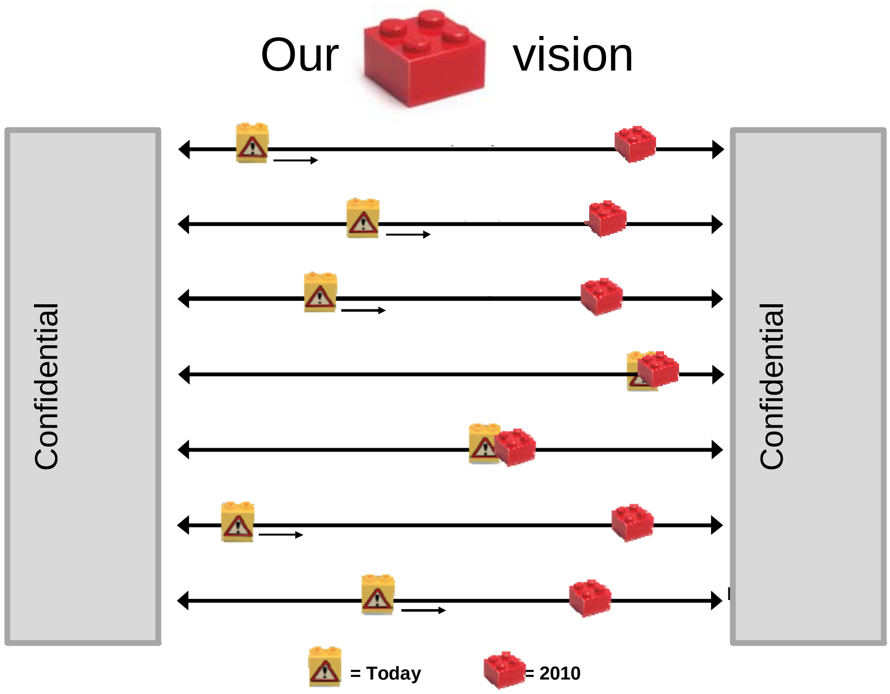


Figure 42: Strategic aims similar to the Shared Vision document. This slide shows how to illustrate aims by defining where we are today (2006) and where we will be in 2008 for a number of element characteristics (2010 novelties are developed in 2008).

Another change from 2005 to 2006 was that a new strategy document from top management arrived between the two years. This strategy document, the *Shared Vision* which has already been mentioned, details in great depth where the company is now and where it is going. For a set of toy characteristics and properties (e.g. low-age/high-age, time-consuming/instant-gratification, peaceful/conflict) assessments were given on what the current status was and what it should be in 2008 relative to the competition (See Figure 42).

The relation of the *Shared Vision* and the platform projects was discussed in several workshops within the DesignLab unit. Company strategies were broken down into well-defined strategies for the elements and building systems. This process not only made it much easier to align the future platforms with the company strategy, it also led directly to several platform candidates, which aimed at bringing specific segments or the entire assortment towards the 2008 situation in the *Shared Vision* document.

Finally, a change in the organization brought the product platform creation fully into the hands of the DesignLab organizational unit (described in section 4.1.2). Packaging and production equipment platforms as well as technology-heavy product platforms were separated from the product platforms, and as a result the product platforms became a more distinct group which easier could be synchronized with the novelty projects.

Since DesignLab was already managing and maintaining the element assortment and already had several channels of communication open to the designers in the novelty projects and to management, the difficulties with all communication and alignment with these stakeholders were also greatly reduced.

Platform Opportunities

Platform Opportunities	Illustration	Stakeholders () = not confirmed		Driver (Hovedansvarlig er understreget)	Prioritering
New cover up elements		Technic, Racers, Mindstorms, (City, Play Theme models)		JH/SS	OK
Functional element blocks		Technic,, Racers, (Creator, PLay Theme, City)		SS/JH	OK
New animal building system		Play Theme IP, Play Theme		JR/EL	Skal gøres skarpere
Wall/house element family		City, Belville (Play Themes, Creator)		JR/EL	OK
Frame element family		Technic (Exo Force, Bionicle, Play Themes)		TSJ	Ude
DUPLO vehicle family		Preschool		JR	Afventer øko. oplæg
Revision of Joints & hinges		Exo Force, Bionicle, Spirit Warriors, (Creator, Play Themes IP)		EL	Skal gøres skarpere
Shape element family		Play Themes, Racers, Design Lab. (Play Themes IP, City, Creator)		JR/JH	**Relateres til nedenstå.**
Add universal elements		Play Themes, Play Themes IP, City, Creator, Technic, Racers, (Preschool)		TSJ	
Mini figure Accessories		Play Themes IP, Creator, Creative Building, Belville, City Design Lab. (Play Theme)		OFN/JR	OK **kort præs.**

ELEMENT PLATFORM 2008+ 1

Figure 43: Platform candidates in 2006 (for 2008 products). The two candidates with red rows were discarded even before the P1 milestone: One because it could be incorporated into another platform; and one because the concept was too weak.

The tasks of the project team were to generate the actual platform candidates as before, but now the same people would be responsible for the following development process of all the individual projects, and the emphasis on development of the process was almost completely removed.

This time the emphasis was on creating good platform concepts which were aligned with designer requirements and company strategy.

Many workshops were held where designers were asked about the requirements and needs for coming novelty projects. Reducing the number of platform candidates meant choosing only those candidates, which had the correct timing (i.e. could be aligned with the novelty projects).

The gross list of platform candidates this year consisted of eight projects (See Figure 43), which all held good possibilities. The list of confirmed (positive) stakeholders for each of these projects was also much longer than the previous year.

The eight candidates were taken through the same set of milestones as the previous year. But having far less candidates than the previous year, it was possible to put more detail into each of the concepts. Stakeholder meetings were continuously held so that the concepts did not evolve into something which would not be used in the novelty projects.

The presentations at the milestones were briefer than the previous year, though. More effort was put into describing the concept of the individual candidates and listing the stakeholders. It also quickly became apparent that even though several platform projects would not show high economic benefits (in the short run at least) the clear connection to the *Shared Vision* document sold the projects.

The presentation material itself often consisted of only one primary slide (presented as an A1-size poster) and a number of additional slides, which could be shown if specific questions related to any material not on the primary slide.

The great benefit of this presentation technique was that we were able to quickly communicate the main message to a large number of people in many short meetings. Management and stakeholders were now able to quickly assess and remember each potential project.

Another benefit of this presentation technique was that management was made to take decisions on a relatively high concept level, which is really what is necessary. Cost/benefit analyses and risk assessments often speak for themselves. The fundamental question is if a platform concept is aligned with the company strategy and meets the requirements of tomorrow.

The result of these changes compared with the previous year was remarkable. Management rejected no projects, but the platform team closed four projects along the way to further increase focus on the best platform opportunities.

At least two of the temporarily closed projects from 2006 would reappear in the 2007 list of platform candidates.

Screening 2007 project

Following the success of the 2006 process, the 2007 project remained largely unchanged.

One improvement of the project was an increased emphasis on projects based on the Technic and Duplo building systems, since all of the approved projects of the 2006 process had been based on the LEGO System building system.

The initial workshops were very creative and effective. Since many product designers had participated before and seen the positive consequences in 2006, it was easier to gain their acceptance and make them contribute to the concept creation.

The gross list of platform candidates this year was only 7 projects (see Figure 44) to reduce the workload since many concurrent activities were happening in DesignLab.

The candidates were taken through the same set of milestones and presentation templates as the previous year. The process was pushed two weeks backwards to ensure that components were ready for the novelty projects, and the presentation templates were changes slightly.

In the end four projects remained as in 2006, and the process would continue in 2008.

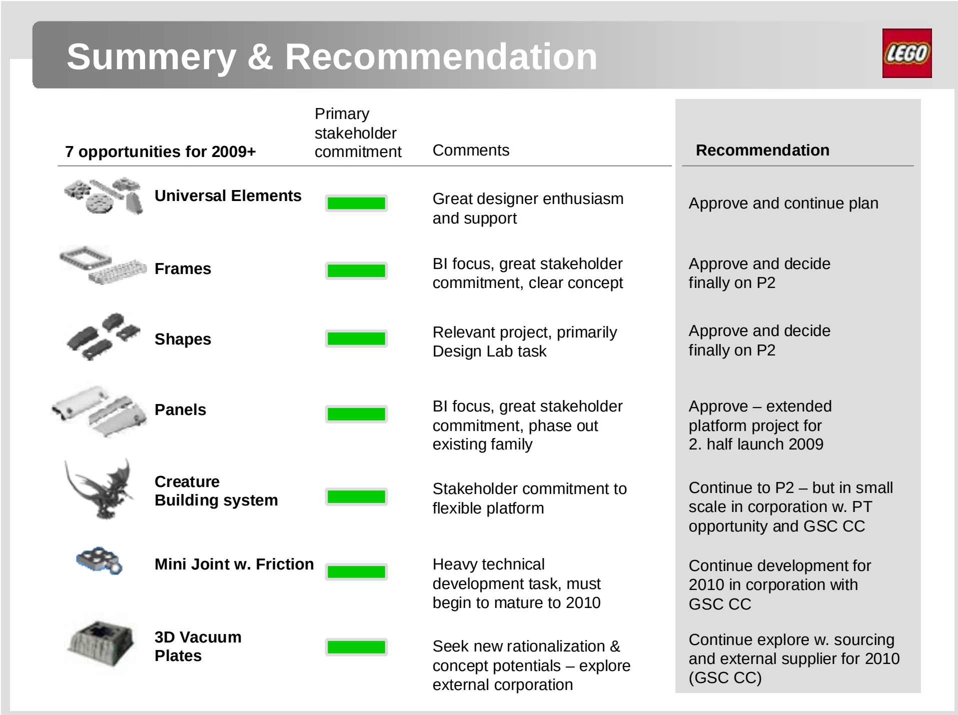


Figure 44: Platform candidates in 2007 (for 2009 products). The bottom three candidates were discarded, primarily because they represented a need more than a ready concept.

Platform projects under the screening projects

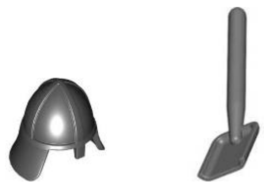
A number of different platform projects were initiated in the screening processes described in the above. Many projects were discarded along the way, but some persevered even though these were very unalike.

Projects	Time frame	Sub-assortment	Architecture	Components	Results
Power Functions	2005-2007	Approx. 10 components replaced by 15 new components	Strict: All components are predefined, can be extended later	Existing components are replaced, new ones are added	New product opportunities, improved quality

Figure 45: The Power Functions project summary. Even though the project affected only a limited number of components, these components contain electronics and are therefore much more complex than traditional LEGO bricks.

The *Power Functions* project of 2005 described in Figure 45 replaced a number of independent components. Motors, wires, battery-boxes, sensors, and similar components were very expensive and not very successful. To improve on the chances of evolving the whole product assortment to a higher electronic level, a new assortment of electronic components was badly needed.

The project succeeded in developing a whole range of components, which are only recently appearing in LEGO products. These components are widely applicable and of a high quality, and the project itself has been a great success, especially for the participants who succeeded in carrying the project through a long initial process with many investments and much opposition.



Projects	Time frame	Sub-assortment	Architecture	Components	Results
Minifigure Accessories Project	2006	Approx. 300 components evaluated	Loose: Components are evaluated and classified based on application potential	Some existing components are eliminated	Reduced maintenance

Figure 46: The Minifigure Accessories project summary. The sub-assortment was reorganized and evaluated, but the project failed in creating a lasting classification method and was therefore not successfully implemented in the architecture.

The *Minifigure Accessories* project of 2006 described in Figure 46 was a very different project compared to earlier and concurrent projects. The purpose of this project was to create a basis for clean-up, additions, and redesign of the components and give an overview in a part of the assortment, which had very few rules and guidelines (i.e. it was not described in the architecture).

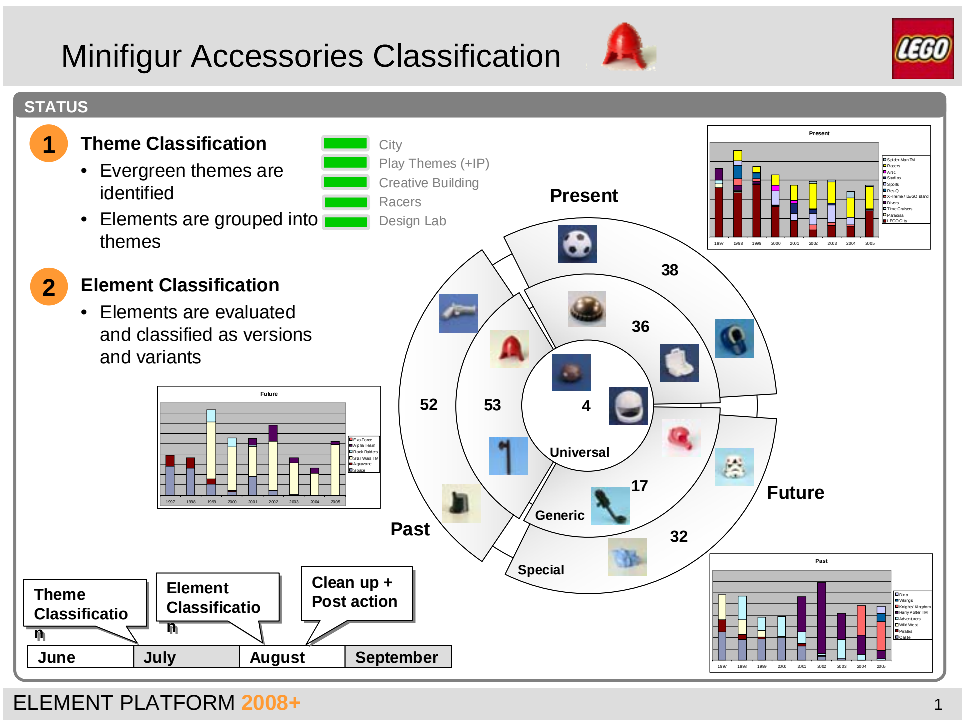
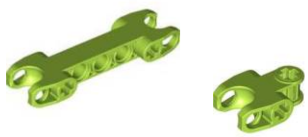


Figure 47: Minifigure Accessories project. The different product ranges (i.e. themes) could be divided into either the present, past, and future categories. Within each of these categories one theme was categorized as evergreen: City, Castle, and Space. Other themes were classified as temporary. Components (refered to as elements) were assigned to each of these themes and classified universal, generic, and special accordingly.

Minifigure Accessories components are one of the biggest sub-groups in the assortment, and most of these components are theme-based. Using the general classification guide for LEGO bricks they should all be classified special, but since some of these components are in fact used permanently in many different products and are produced in high volume it is not correct to classify them as special. The project team created a new classification method for this sub-assortment based on application of the components. Products ranges were defined as evergreen or temporary and components were assigned to specific themes according to where they were applied the most (see Figure 47).

The project succeeded in creating an overview, which was used for clean-up, but failed in making a clear permanent classification method. The long-term benefits and purpose of the project remained unclear for many, who easily mistook the project for another clean-up project.



Projects	Time frame	Sub-assortment	Architecture	Components	Results
Ball Joint Project	2006	Approx. 30 components revised	Medium: Interfaces are updated	Existing components are replaced	Improved quality

Figure 48: The Ball Joint project summary. This project was based on a technical quality problem (it was too hard to meet specified tolerances), but because the solution would affect a large number of related components (both existing and future) it was still considered a platform project.

The *Ball Joint* project of 2006 described in Figure 48 was a very different project from those described in the above. The main purpose of the project was to improve the quality of the ball joint connections. Ball joints are the primary connectors for the successful Bionicle product range, and therefore an essential component for the company. However, the geometry of the ball joint is difficult to produce, and production samples showed big variance from the specified tolerances (see also Figure 49).

The project team analyzed the ball joint and came up with a number of improvements, which would improve the quality of the ball joint and lower the age requirement for children playing with affected LEGO products. The project was one of the more cross-organizational projects since coming up with new solutions required in-depth knowledge of production techniques as well as product function requirements.

In the end a number of different components, which had ball joint interfaces were modified and improved and the project was very successful.

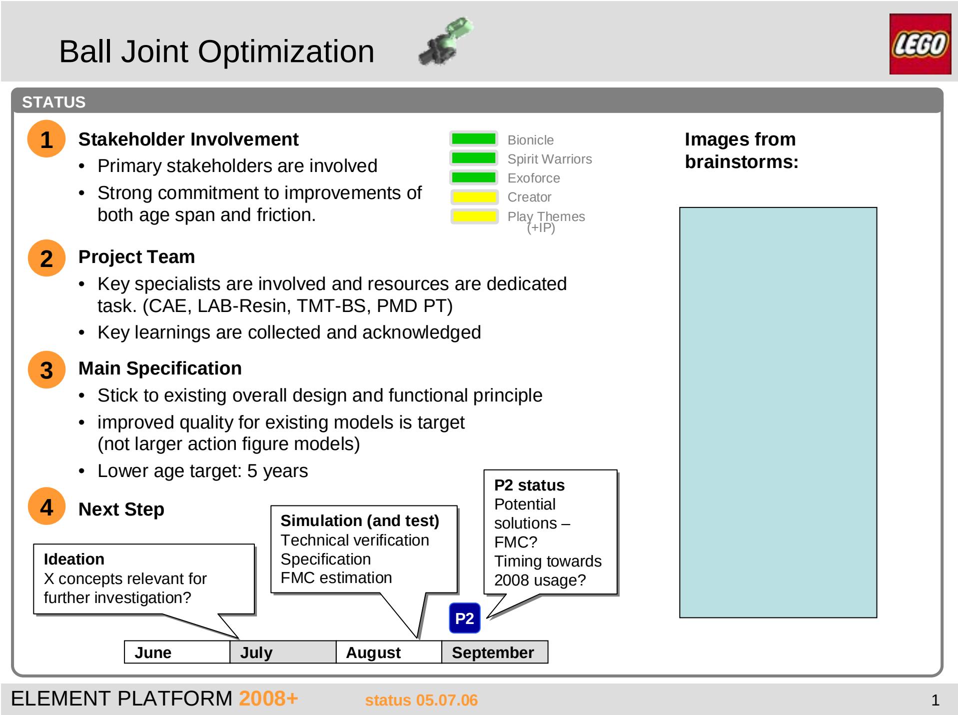
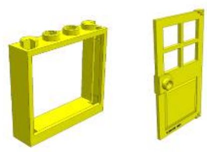


Figure 49: The Ball Joint project. Stakeholders and project members were very important in this project, since it was truly cross-organizational. The project aim was to improve the quality (i.e. improve the static friction and make it the same for all components) and lower the age target by making it easier to assemble.



Projects	Time frame	Sub-assortment	Architecture	Components	Results
Walls & Windows Project	2006-2007	Approx. 50 components reduced to 35 components	Medium: Core components are predefined, guides for extensions	Best existing components survive, others are replaced, new ones are added	Reduced maintenance, Improved quality

Figure 50: The Walls and Windows project summary. This project resembled the Wheels project, but did not make the architecture finite. New components can be added in the future, although the must comply to the rules specified in the project.

The *Walls and Windows* project described in Figure 50 of the 2006 process resembled the *Wheels* project in some aspects. The purpose of the project was to redefine and redesign all the large structural bricks typically used for house or castle building in LEGO products. As in the *Wheels* project the basic assumption was that *Walls and Windows* components themselves should be more structural building bricks (i.e. universal bricks) than product-selling bricks (i.e. special bricks), and therefore they should be standardized to a higher degree. Also many existing components were not utilized because they were too expensive.

The project team designed a number of components which would replace or supplement existing components, but many small product designer disputes over specific concepts made it necessary to continuously reiterate and the project was postponed several times.

The end result however was still a small success though, since the *Walls and Windows* assortment had been reduced slightly and improved significantly, so that all product developers agreed that it had generally become better.

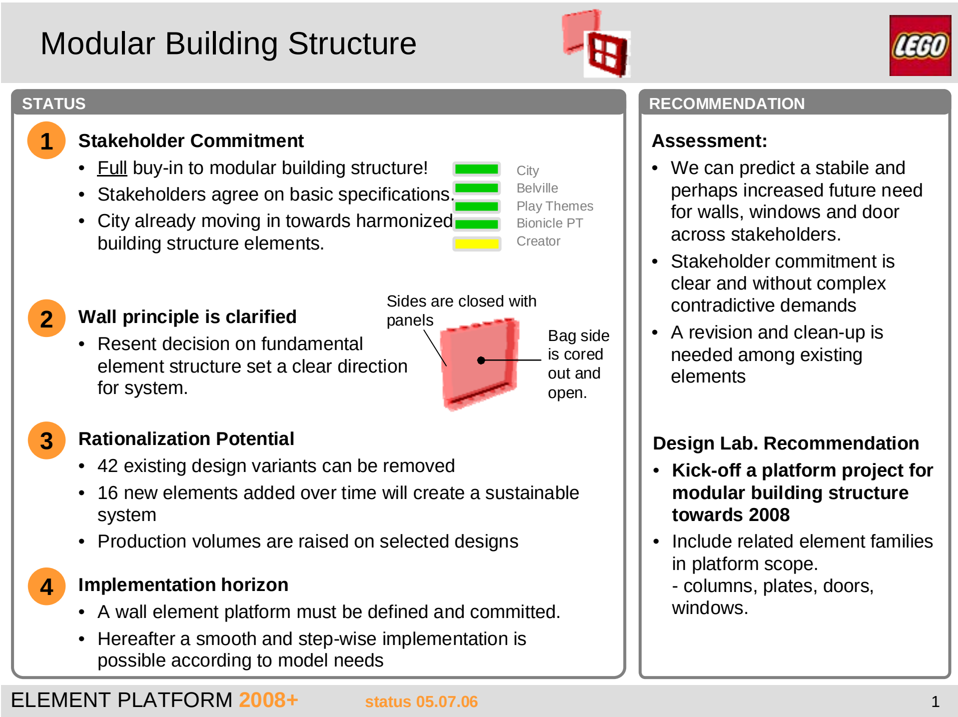
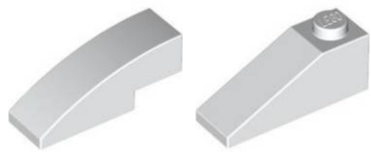


Figure 51: The Walls and Windows project. The project was founded on a new concept for wall components based on an open backside instead of an open bottom as in the traditional LEGO bricks.



Projects	Time frame	Sub-assortment	Architecture	Components	Results
Shapes Project	2007	Approx. 50 components evaluated	Medium: Components are classified based on compliance w. rules	No changes	-

Figure 52: The Shapes project summary. This project resembled the Minifigure Accessories project, but this time architectural rules were defined. The sub-assortment was reorganized and evaluated, but again the project failed in creating a lasting classification method and was therefore not successfully implemented in the architecture.

The *Shapes* project of 2007 described in Figure 52 was another attempt to launch a classification project like the *Minifigure Accessories* project. Based on the unfortunate experiences from that project, the *Shapes* project was defined more as a guideline project, with emphasis on how to add future components and how to classify the components.

Like the *Minifigure Accessories* project, *Shapes* was another section of the component assortment, which was not covered sufficiently by the component architecture. *Shapes* components were generally considered universal because of their appearance and broad application potential, but many equivalent *Shapes* components existed and production volumes differed greatly.

The *Shapes* project defined a new overview of the *Shapes* components (see Figure 53), but again the project failed in creating a lasting effect. Conclusions from the project were that the proper way of implementing new classification methods (i.e. architecture extensions) has yet to be designed, and that projects, which focus on architectures and not components are overlooked and intangible at the time being.

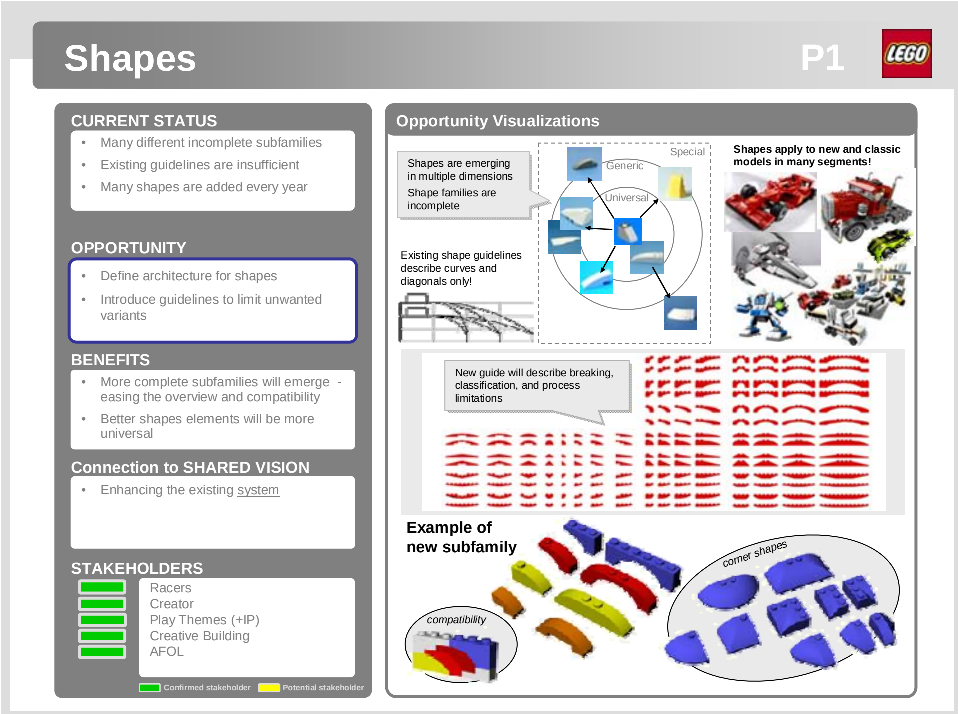
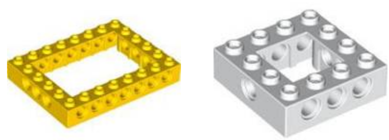


Figure 53: The Shapes project. Shape components were categorized after dimensions and shape-type, but the recommendations on classifications did not receive commitment, because product designers disagreed on which components was more applicable.



Projects	Time frame	Sub-assortment	Architecture	Components	Results
Frames Project	2007	3 new components created	Medium: Core components are predefined, guides for extensions	A few new components are created	New product opportunities

Figure 54: The Frames project summary. This project created new components, which could lead to new product opportunities and could improve the perceived building quality. But due to product designer reluctance and uncertainty only three components were approved and the project became less of a success.

The *Frames* project of 2007 described in Figure 54 was originally also thought to create an entire new assortment of frames for LEGO System and for LEGO Technic (see Figure 55). There was no existing sub-assortment to be replaced, and this became a problem since there were no prior quantities to build a business case upon, and product designers were not familiar with the components and therefore did not use them. As the project evolved it was gradually reduced in scope, and the outcome (two new frame components for LEGO Technic and one for LEGO System) was not very impressive and scarcely enough to be called an assortment.

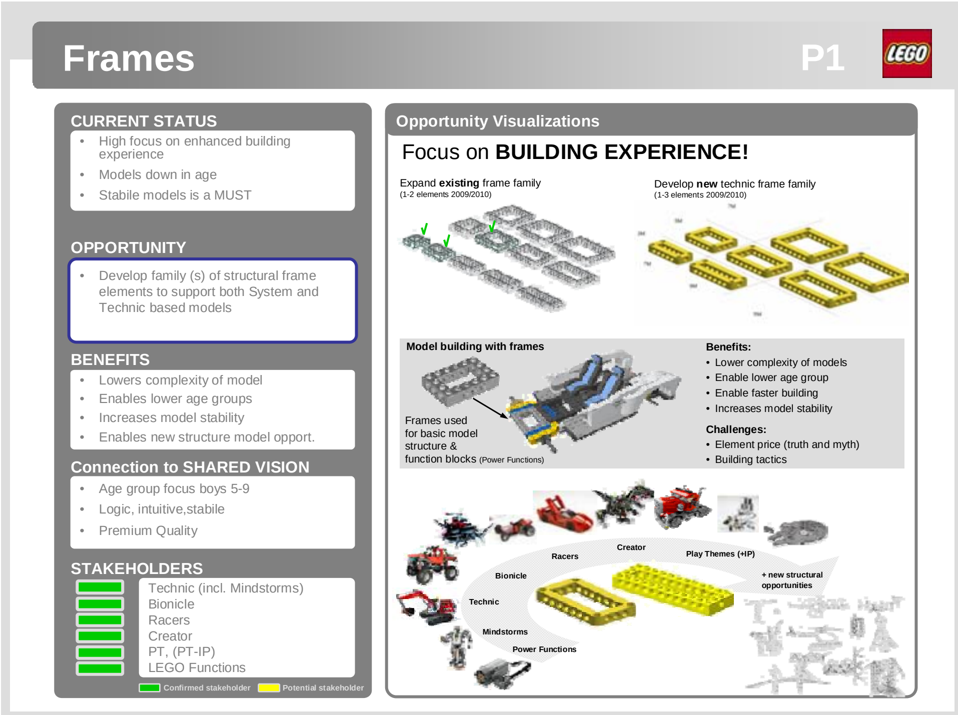
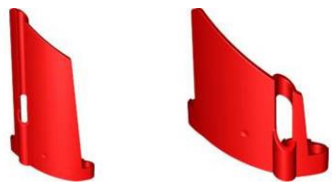


Figure 55: The Frames project. Frame components were thought to appear in to variations: one for LEGO Technic and one for LEGO System. These frames would support the chassis-building on various vehicles and robots in many sizes.



Projects	Time frame	Sub-assortment	Architecture	Components	Results
Panel Project	2007	Approx. 10 components replaced by 15 new components	Strict: All components are predefined	Existing components are replaced	Improved quality

Figure 56: The Panels project summary. The new Panels components replaces a previous assortment of panels for LEGO Technic completely. The new assortment out-performs the old one significantly.

The *Panels* project of 2007 described in Figure 56 was even more radical than the *Wheels* and *Walls and Windows* projects. The existing sub-assortment of panels for the Technic building system was expensive, problematic to produce, and hard for children to assemble. As a consequence it was decided that it should be completely replaced by a new and better assortment.

The new system was designed and agreed upon relatively quickly. As in all Technic projects it was relatively easier to gain consensus due to fewer stakeholders compared to LEGO System projects. The result of this project left the participants with a feeling of “Why didn’t we do this before”, since the new *Panel* assortment was much better than the previous one.

The new *Panel* assortment is more modular, more generic in appearance (i.e. more widely applicable) and components are easier to distinguish, which is good for both children and production (see Figure 57).

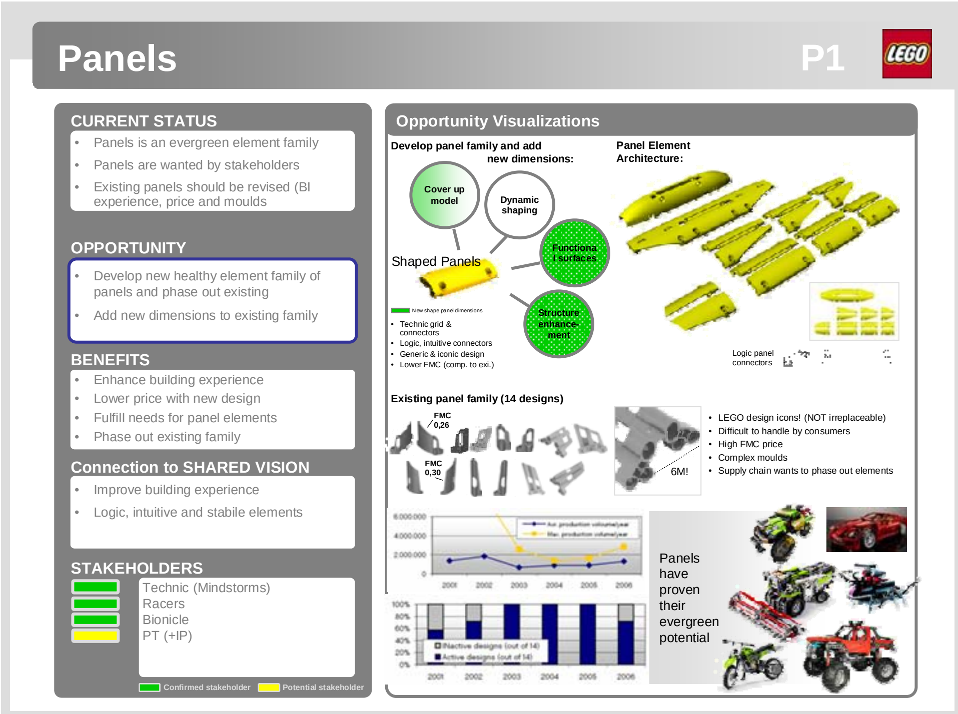


Figure 57: The Panels project. The new Panels (top right) are more modular and generic appearance than the old ones (middle left) while still maintaining a distinct design look.

Universal Elements platform projects

The *Universal Elements* projects are some of the most successful platform projects, which have been developed as part the platform screening processes since 2006. The motivation for these projects has been the desire of separating the development of universal components from the novelty projects.

The need for a universal component should not be based upon any single novelty project anyway, and within the novelty projects there is not the overview or the dedicated time, which is required for the creation of a good universal element.

A possible solution to this problem is to divide the element frame into two separate parts as shown in Figure 58. One part will be developed in the novelty projects as usual. The other part will be dedicated to creating universal components. This is what the *Universal Element* platform projects are all about. In these platform projects it is proposed to develop a number of universal components outside of the novelty projects. If these new universal elements are a success (i.e. they are utilized in several different novelty projects and produced in relatively high numbers), the project will continue, and a new sets of universal components will be proposed.

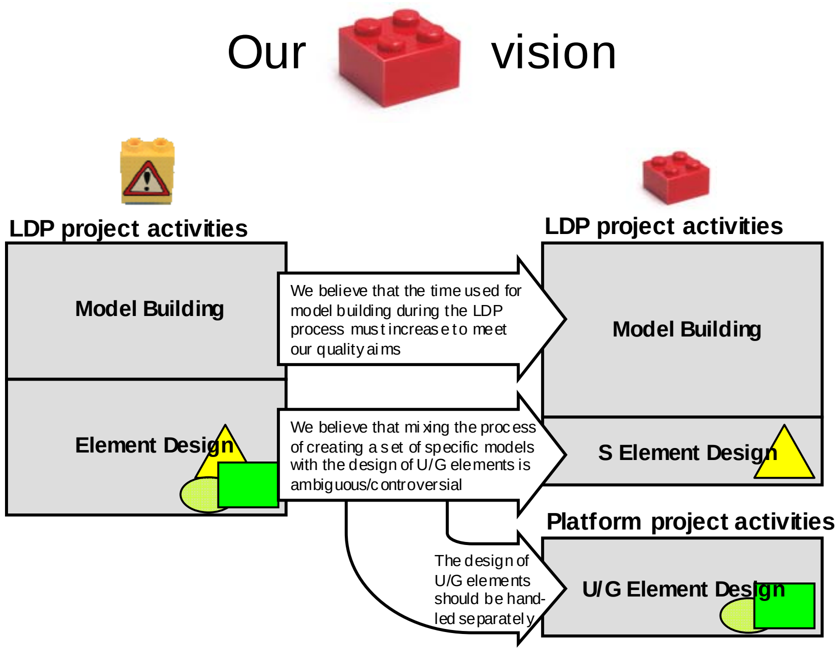


Figure 58: The split between in-project development of shared components and separate development of shared components.

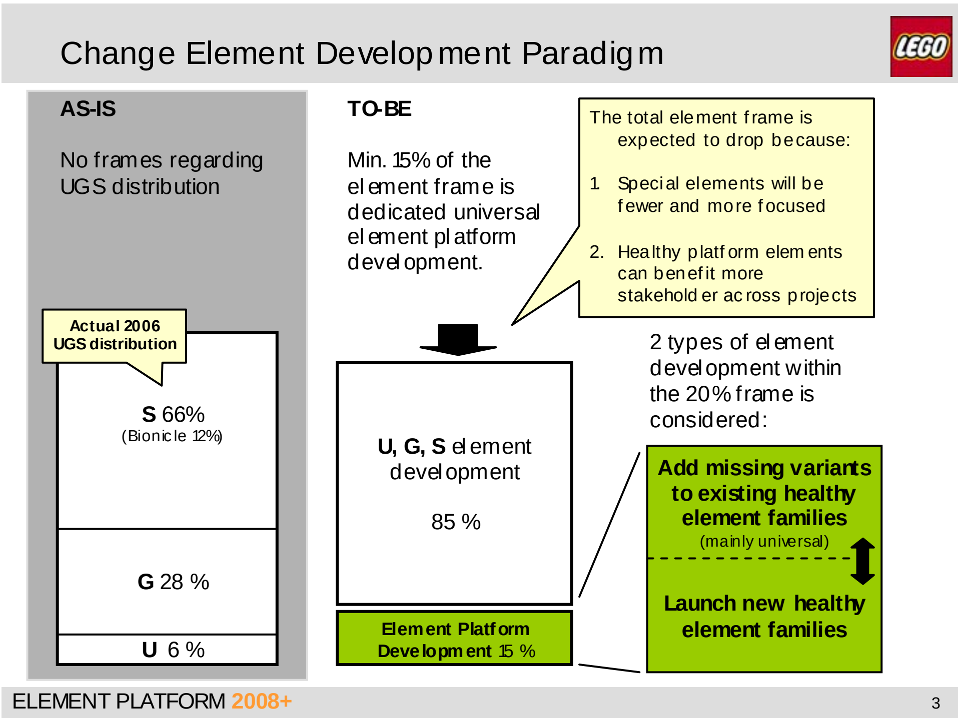


Figure 59: Component development at LEGO. Of the 15% components that are developed in platform projects, some are part of the Universal Elements projects (add missing variants) and others are part of the remaining platform projects, where they are developed in component families. The 15% will likely decrease in time.

This split will effect the composition of the component assortment. It will increase the annual growth of universal components, and (because component frames will be lowered for novelty projects) it will decrease the annual growth of special components. As a result we assume that the quality of the remaining special components will rise, partly because more time can be dedicated to these components, and partly because only those components with the best purpose will be created. The quality of the universal components will also rise, because they will now be based on broad application potential and not special needs.

The first iteration of the *Universal Elements* project in 2006 succeeded in creating 10 new universal components and the second one in 2007 succeeded in creating 5. These components were elected by involving all product designers in the company in sketching new universal components and afterwards voting for the best ones. The elected components are therefore various unrelated components compared with the highly-related components developed in the other platform projects.

Generic components may in time also be taken out of the novelty projects (as suggested in Figure 58), but these generic components should not be brought together with the universal components, or the concept of universal components may be undermined. This was tried in 2007 and resulted in a lower number of components than the previous year, because a number of generic components were taken out of the process in the later stages – much to the frustration of product designers who had anticipated using these components.

A possible solution to this problem is the renaming of the project into something not containing the word universal, but this solution is risky, because management might still perceive the project as a dangerous loosening of the classification – dangerous since novelty project economy is related to classification (described in section 4.1.2).

So far the components being developed in the platform projects add up to about 15% of the total number of components being developed every year as shown in Figure 59. It is however not likely, that this proportion will be permanent. Universal components are by definition almost permanent and since it is the current LEGO strategy to stabilize the total brick number, universal component development should decrease so that it only replaces those few that are discarded.

4.2. The Technology Center at Grundfos Company

Grundfos Company is a world-leading producer of pumps of all sizes (see Figure 60). The Technology Center (TC) at Grundfos is responsible for various kinds of support for the main business, they design and produce the equipment needed for the production of pumps, which includes molds, stamping equipment, robot cells, test beds, etc.

Figure 60: Typical Grundfos products: Pumps with pump-housing.

Grundfos TC has a long tradition of designing and producing this equipment and has acquired a certain know-how of how to produce this and what the technical possibilities and capabilities are. However, as all of this equipment is considered secondary compared to the main manufacturing of pumps, it has been difficult to standardize anything, since the equipment is designed based on specific needs, which changes with new generations of pumps.

To remedy this, a number of platform projects have been initiated in TC. The purpose of these projects is to capitalize on the know-how and create preferred solutions, which can be utilized as platforms, and which showcases the benefits of complying with standard solutions in terms of significantly reduced order lead-time, costs, and risks.

This section of the thesis is dedicated to describing the two platform projects at Grundfos TC listed in Figure 61, as well as the background and setting they have been introduced into.

Projects	Time frame	Description	Researcher participation
Molding Equipment Project	2006	Cleanup, guidelines, and revision of molding equipment	Full participation
Test Bed Project	2008	Cleanup, guidelines and revision of test bed assortment	Full participation

Figure 61: Grundfos TC platform projects documented in this thesis. More platform projects are scheduled in the coming years.

4.2.1. The molding equipment platform project

Molds are designed and produced for various products in Grundfos TC. In Figure 62 a selection of molded products are shown. Common for these products is that they can be put into water (or sometimes also other fluids) circulations systems, they have a number of cavities or extrusions for mounting of small devices or connections to other parts of the system, and they are made of a glass-fiber-component plastic material.

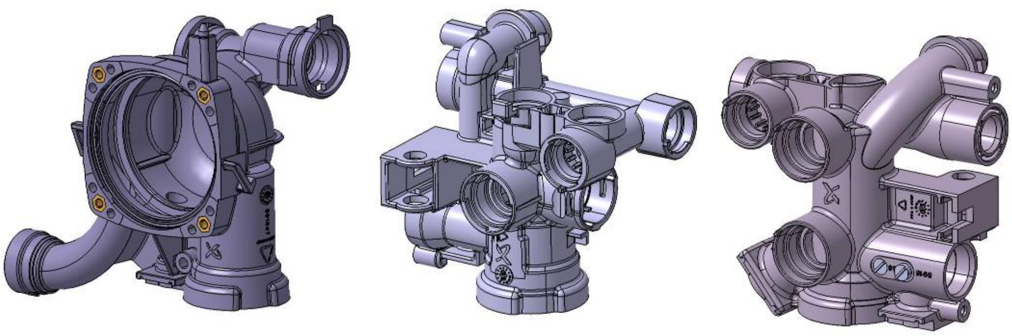


Figure 62: Grundfos TC molded products (from left to right): A pump-house, a flow group, and a return group.

As can be seen from the products in Figure 62, the molds must be relatively complex with a large amount of guides and moving parts. The overall structure of the products differs a lot from product to product (also between different pump-houses) and the positions of the various extrusions are new every time. The products are roughly the same size though (approx. 15cm square) and many connections are alike although few are identical.

A common feature of the pump-houses is the house-cavity itself, which is created by a collapsible core (see Figure 63), which retracts in two steps when the mold opens. Grundfos

TC reuses the same concept for collapsible cores, but variations in the cavity-shape and positioning of mold ejectors prevent component reuse of any kind.

New molds are ordered from Grundfos TC, whenever a new pump has been designed and must be put into production. Molds are rarely reordered, but are instead repaired onsite.

The burning platform

Following the globalization of the pump manufacturing industry, Grundfos is increasingly looking for new ways of improving their products or reducing the costs of their products. Internally departments are being benchmarked against outsourcing alternatives. This also applies for Grundfos TC, and in the later years it has become apparent, that molding equipment could perhaps be produced faster and cheaper.

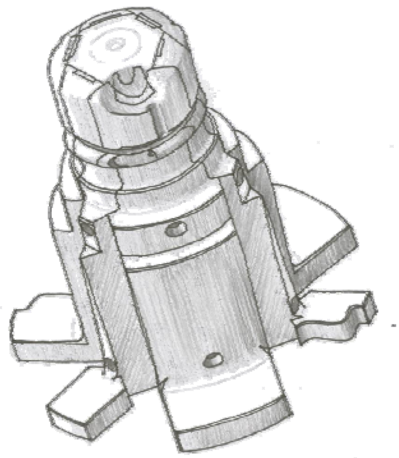


Figure 63: Sketch of a collapsible core, which forms the large inside-cavity of a pump-house.

Grundfos production sites are increasingly questioning why it must take so long to design and produce the molds for pumps, while production is waiting for them to arrive. The molds from TC are generally of a very high quality, but comparative mold manufacturers are gaining on them, and their order lead-times are significantly lower.

Another issue is accuracy of estimates. Grundfos TC gives estimates on order lead-time and costs before beginning a new order, but these estimates are often very rough. Very fine tolerances require some parts to be measured and adjusted several times – greatly increasing the costs and time of manufacturing, but even more important are the technical uncertainties, which are created by constantly introducing new kinds and sizes of connections and their relative placements.

The burning platform is getting warmer. Mold manufacturing in Grundfos TC must either improve significantly or face outsourcing.

The collapsible core platform

It was initially decided in the platform project, that the similarities, which could later be used as a platform, would have to be the connections. The connections ordered by the customers are mostly very different, but some are similar from one mold to another. Some key-employees from TC also believed that a significant cost difference between a standard connector and a customized connector could change the customer preference in some cases.

The obvious choice for a standard component for the pump-house molds is the collapsible core, which create the large internal cavity in the pump-houses. Collapsible cores are not identical because the geometry of the pump-houses differs, but conceptually they consist of the same sub-components with roughly the same outer dimensions.

A few other connections have been reused conceptually in different molds, and can therefore perhaps be standardized and used as platform components. One type of connection is a bayonet connection, which occur in two different sizes (15mm and 18mm diameter), and another is a screw thread connection (M10).

The mentioned connections are shown in Figure 64. The collapsible core is designed in three conceptually different variants: A tall one, a short one (shown in Figure 64), and one with an extra component placed on the top (called a hat). The bayonet is designed in two variants: One which is perpendicular to the mold injection and one which is parallel to the mold injection, each of these occur in two different sizes as mentioned in the above.

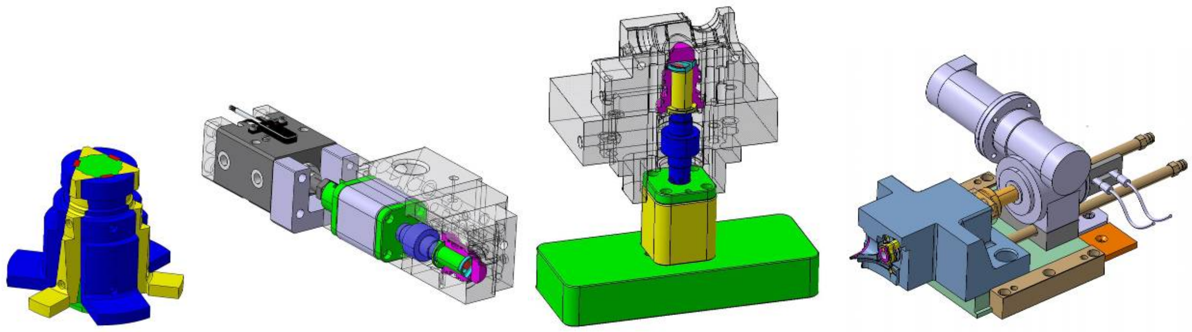


Figure 64: Mold inserts, which correspond to the most common types of connections in the molds at Grundfos TC. From left: Collapsible core (short variant), bayonet (perpendicular variant), bayonet (parallel variant), and screw tread. The collapsible core is scaled down, and is in reality much bigger than the other inserts.

The shapes of the mold inserts are never completely identical, but customized according to specific needs in the products, and it is at the moment impossible to change this fact.

An early and very important recognition in the projects was however, that a specific emerging processing technology (thread cutting) would enable this customization of the mold insert to be carried out in one final process step. This meant that the mold inserts could be produced generically to a certain point following identical process steps, where after a customized process would finalize the insert.

Much of the order lead-time was made up by the early process steps, but since they would become generic, they could be started as soon as an order was given or even earlier and then put on stock. Meanwhile the final specifications and drawings could be designed. This approach is shown in Figure 65.

In the top preparation phase, the generic collapsible core is chosen and manufactured, it has already been designed, all drawings are done, and all processes well-known and predefined. This phase must be carried out like an automated process started by the push on a button, it requires no decision-making or creativity.

In the bottom, execution phase the customized collapsible core is designed and manufactured based on the generic collapsible core. The drawings and processes show the modifications applied to generic collapsible core, as if this generic collapsible core had been a raw material part or a standard machine part (like an M6-screw). This phase must be carried out by designers, who know the limitations and possibilities of the platform, it requires some decision-making and creativity but much of this can be described in guides.

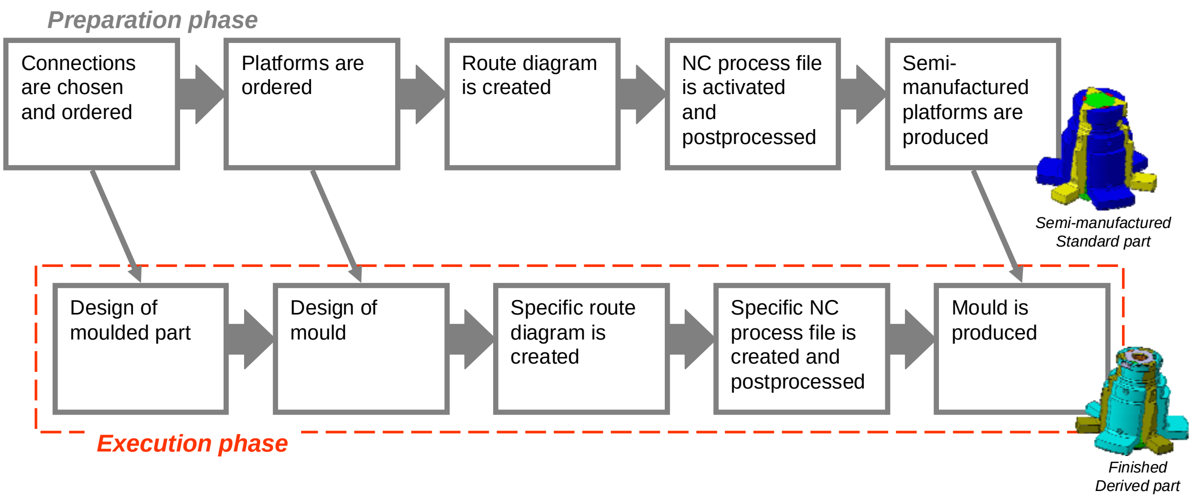


Figure 65: The split of the collapsible core manufacturing process. The mold design and production is now split into a preparation phase and an execution phase.

The truly experienced designers are hereafter needed for deciding which platform components match a new order and for designing new platform components, which can be used in future orders.

Implementation

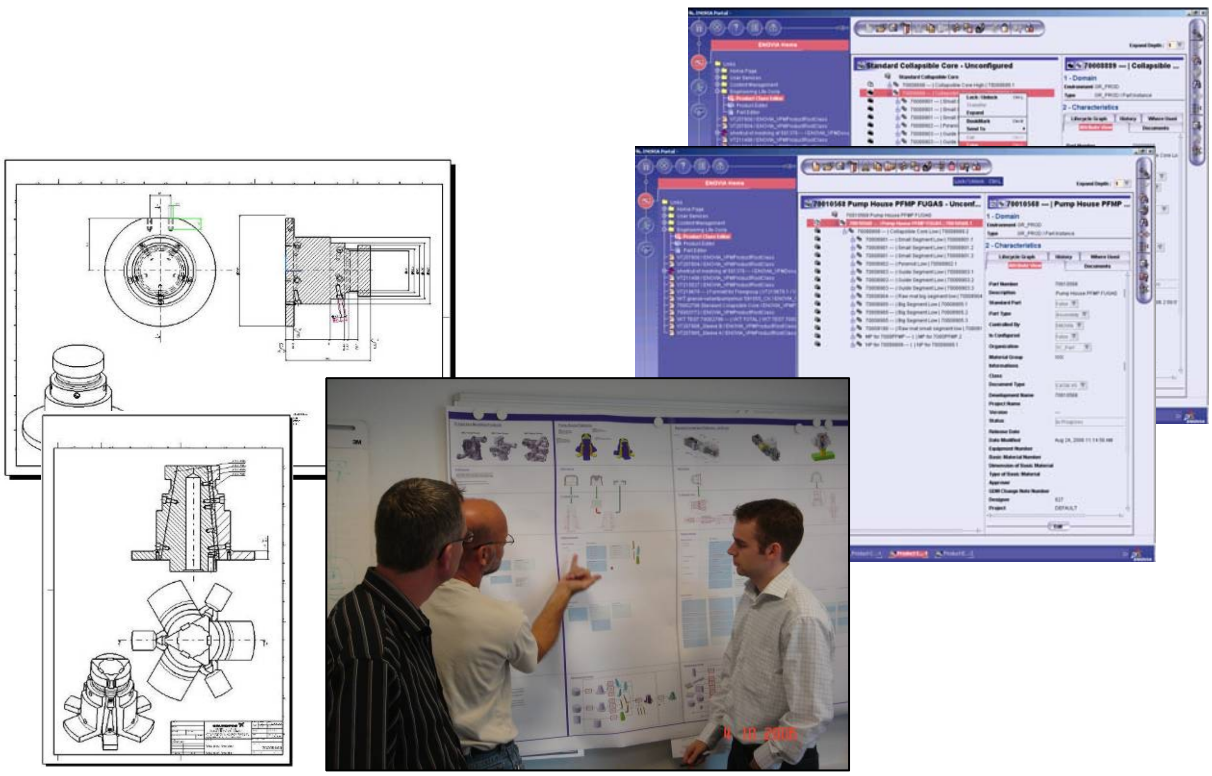


Figure 66: Left: Drawings of the two collapsible core platform variants (i.e. short and tall). Right: The PDM structure of the collapsible core platforms. Middle: Shared discussions of the platform structure on a PFMP board.

The generic platform components were redesigned based on best original components and the drawings and product structure were saved in special location in the PDM and ERP systems reserved for reusable platform components (created as part of the project). As in many other projects it did however not suffice to make platform components available. Special attention was given to anchoring and implementing the use of platform components in the employees work routines.

The design (i.e. the modifications based on previous designs) of the platform components was conducted through a series of shared discussions and workshops involving the primary future users of the platforms (see Figure 66). Furthermore a simulation of a full process from order receipt to assembly of a collapsible core was carried out in a full-day workshop, which involved representatives from pump-house design, mold design, planning, NC programming, and production.

The full implementation of the collapsible core was very successful, and all collapsible cores created at TC must now be platform-based. The other platforms based on bayonet or screw thread connections has however not been completed, partly because of too little demand on these components, and partly because they are more complex.

The case of the molding equipment platform has previously been described in [Mortensen et al. 2008].

4.2.2. The test bed platform project

Test beds are designed and produced for two purposes at Grundfos. First and foremost all pumps are tested as a final step in the product process. Secondly customers can always bring Grundfos pumps to Grundfos service centers to have them tested.

Figure 67: A medium-sized test bed for end-suction and inline pumps.

Test beds exist in a number of different variants for different kinds of pumps, ranging from small (can be fitted to a table), medium (can be fitted into one container), and large (requires many containers and more than 30sqm on location). A medium-sized test bed for end-suction and inline pumps is shown in Figure 67. This is one of the most popular test beds, and was therefore suggested for the platform project.

Test beds are designed for testing the pump characteristics in QH tests as well as static high pressure performance. A number of more specific tests can be applied to special pumps and all pumps are also observed for leaking when put into a test bed.

The burning platform

Grundfos pump factories around the world require a number of testers for the final testing of all produced pumps. Test beds are however by default designed and produced in Denmark by only a few experienced designers. This has led to long delivery times and supply shortage.

Production of pumps cannot commence before test beds are ready, so an increasing number of factories are looking into alternative solutions. Some factories are building test beds themselves and others are hiring local craftsmen to do the job. All of these alternative solutions have to be quality-approved by the few experienced engineers in Denmark though, and there are many examples of unqualified workers building inadequate test beds, which are disapproved.

Additionally a number of Grundfos service centers possess only old test beds, which are inadequate for testing newer pumps. These service centers are potentially interested in acquiring new test beds, but they have a very limited budget.

There is clearly a need for a standardized test bed, which can be customized and supplied quickly. Furthermore if the capacity of the test bed designers could be increased by having standard products, this might prevent more future supply shortages.

The test bed platforms

Compared to the variegated mold structures described in the previous section, the structures of the individual test beds of a single kind is remarkably alike. Although a multitude of optional extra functions and equipment exists, there is a common core, which can be standardized.

The primary step of standardizing the common core of the test beds is to create an appropriate amount of documentation, which will assure the reuse of the components and structure. Until this point Grundfos TC has only documented the mechanical parts and their structure in flow diagrams (see Figure 68), which leave much for the creativity of the craftsmen creating the specific components.

The new test bed platform is documented in 3D CAD, which will enable reuse of this design. As can be seen from Figure 69, the platforms are not complete products, but rather a configuration of different components, which are shared between the members of a test bed family. This configuration is not complete (i.e. pipes do not form a closed circuit), meaning that additional components must be added to make a finalized product.

Three standard configurations have been created: One for service centers (which is very basic), one for production plants, and one with two work stations (i.e. two pumps mounted although not tested simultaneously). Based on this specialization of the three test bed platforms a number of new mechanical layouts were suggested, and the new platform-based assortment of test beds is therefore not only based on the best existing solutions, but it also feature several new improvements.

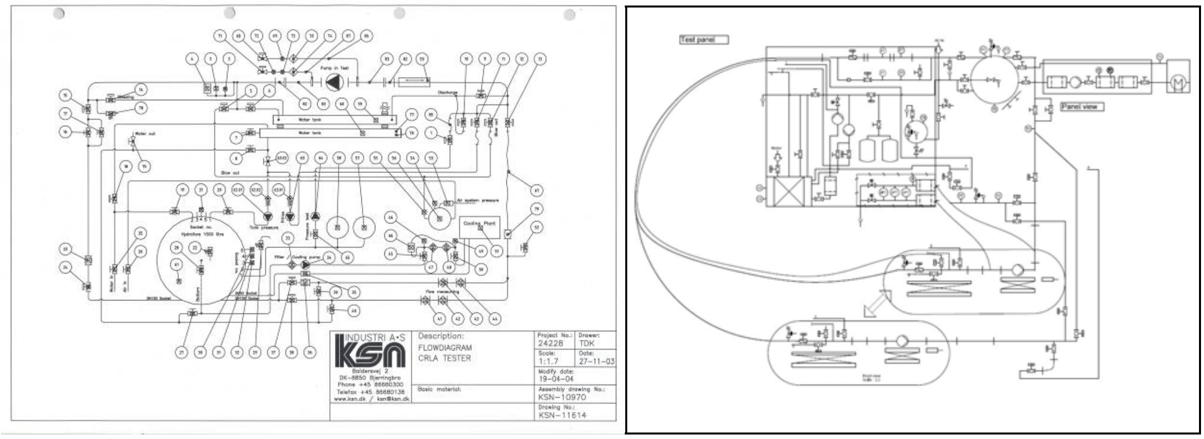


Figure 68: Grundfos test bed flow diagrams. This has been the primary documentation of test beds until now. Left: Flow diagram of the old test bed in Figure 67. Right: Flow diagram of the new test bed platform. The new one is inverted compared to the old one.

In Figure 70 is shown an example of a section of the test bed which features different mechanical solutions for the service center test bed platform and the production plant test bed platform. The previous conceptual solutions for the same sections differed from one test bed to the next, but now two distinct standard concepts with different application potential are available.

Standardizing three test beds leads to a number of consequences. First of all, a lot of design work has been prepared now, when a customer (i.e. a production plant or service center) orders a new test bed, Grundfos TC will no longer start sketching flow diagrams on a blank sheet. Secondly, production of the test bed components can now be outsourced to other suppliers. Finally, the test bed platforms are well documented, so that Grundfos employees with less experience can actually design new test beds based on this platform.

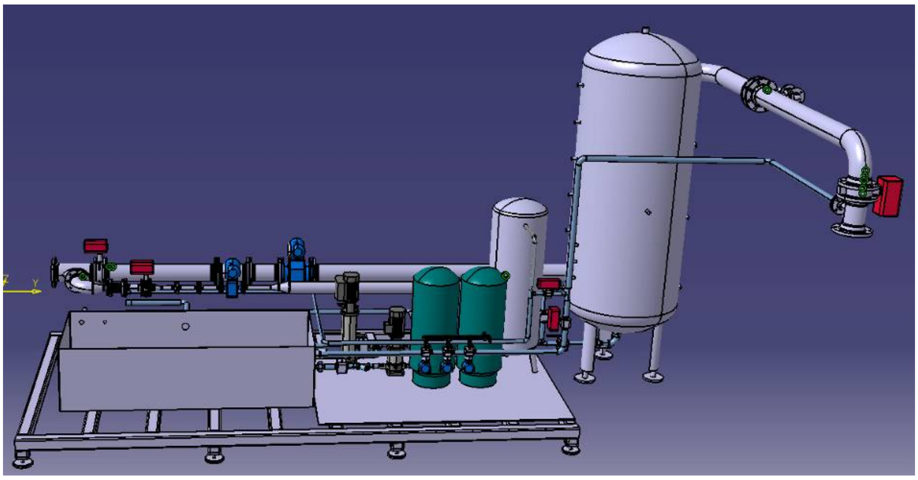


Figure 69: One of the new test bed platforms. The platform has not been finalized in this picture, as the project is still ongoing.

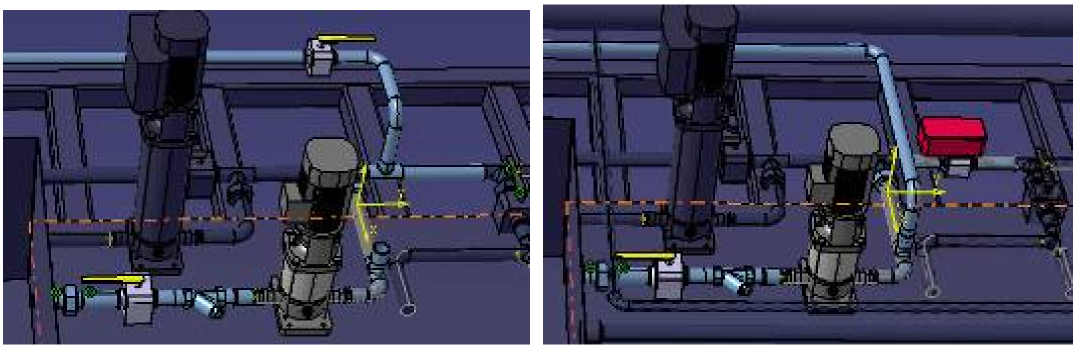


Figure 70: Priming of the test bed is done in two ways. Left: In the simple version, water fills the circuit from the main tank. Right: In the advanced version, water fills the circuit from a waste water tank, which is a faster but more expensive solution.

Implementation

The platform components and test bed architecture were redesigned and saved in a special location in the PDM and ERP systems in the same way as in the mold platform project.

A broad group of experienced and novice test bed designers were involved in the project, so that the experienced designers could incorporate their knowledge in the platform and the novice designers could learn and familiarize themselves with the platform. These novice designers would eventually be building test beds created as configurations of the test bed platform.

The project failed to reduce the costs of new test beds significantly, but this had always been a secondary aim. The primary aims of reducing the order lead-time and increasing the capacity will be achieved when the first platform-based test beds are ordered in late 2008. This assessment is based on the facts, that it will be faster to configure a test bed than to design it, and that it takes less experience to configure it than to design it, and therefore more employees are capable of the task.

4.3. Additional case material

The LEGO and Grundfos cases described in the above are cases which I have participated in (with a few exceptions, which are described in the respective sections because they have led to or have been started because of other described projects).

Besides these cases a number of other cases are documented within my research group, and frequent presentations and discussions of these cases have influenced my general perception of and description of platforms and platform projects.

These cases include other cases from LEGO described in [Munk 2009], cases from Danfoss in [Pedersen 2009] and [Kvist 2009], as well as various cases described in [Harlou 2006].

4.4. Concluding on the cases

The cases described in this section show many perspectives of platform-based product development. Platforms at LEGO and platforms at Grundfos are at first sight not very similar, but neither are individual platforms at LEGO or Grundfos. Sometimes they are sets of common components, sometimes they are rules that define how products must be built.

Sometimes they involve or define only small limited areas of a few products, and sometimes they govern the entire product line in a company.

Common for all of these platforms is that they are not restricted to a single product, and that they employ sharing and design rules to increase the output quality and/or the effectiveness of the product development department. In all of the cases we use the experiences acquired in the past to improve the future products.

Common for all of the platform projects are also the burning platform outset. It is not only LEGO who experience part proliferation. Almost any manufacturing company accumulate components over time [Fisher et al. 1999] – and even companies commonly perceived as efficient (e.g. Toyota) are troubled by this.

LEGO and Grundfos TC strive to implement platform-based product development as an integrated part of their product development. They want product developers to apply and design new platforms as needed, but rely on the product developers themselves to make this turnaround. The product developers however are often not capable of making this turnaround, because they are unskilled in platform-based product development and because they are focused on single-project performance rather than cross-project performance.

In the following sections the cases described in the above will be used to shed insight into how platform-based product development can be made continuous and how it can be synchronized with product development.

5. Platform Projects

The cases presented in the previous section both deal with platform projects. Platform projects are the means to create and modify the platforms in a company, so before trying to answer the research questions presented in section 3 it will be useful to look at platform projects in general.

In this section I will try to describe platform projects in general and classify the different kinds of platform projects, which exists. A natural part of this description is a comparison with traditional product development projects.

This section and the following ones extend the platform paradigm described in section 2. Where the previous platform paradigm descriptions are based primarily on literature and commonly accepted facts, this extension is based primarily on the cases described in the above, although each of the following sections will begin with a summary of various related research.

5.1. Platform projects in the literature

State-of-the-art literature does not deal directly with platform projects. Most authors describe either how to create platforms or what the benefits of having platforms are. Those authors who describe how to create platforms sometimes gives procedures, which more or less can be followed in a platform project. There is however two different starting points for platform projects: Redesign of an existing product family or novel design of a product family [Liedholm 1998], which correspond to a bottom-up approach and a top-down approach [Simpson 2004].

State-of-the-art contributions can generally be categorized as either market-based or technical. Market-based contributions describing platform-based product development typically proposes step-by-step procedures for specifying and designing a common platform (primarily by designing a modular structure) based on the features and variety a given product family should contain [Ulrich 1995, Erixon 1998, Farrell et al. 2003].

Technical contributions on the other hand, focuses intensively on the architecture design based on flow-diagrams (sometimes combined with different mathematical approaches for calculating ideal modular architectures) [Zamirowski et al. 1999, Stone et al. 2000, Dahmus et al. 2001, Hölttä et al. 2003, Van Wie et al. 2003]. In all of these later publications the platform project is considered a redesign of product family, and the product variety is already determined. An extensive review of the various methods is available in [Gershenson et al. 2004].

Assuming an existing platform needs to be modified, many of these methods could of course still be applied for creating an alternative platform. If one module in a modular architecture needs to be further modularized to accommodate for new variety, this module can be considered an independent product with an independent architecture due to the standardized interfaces and then the same methods can be applied. Also, if a new module is to be added to an existing architecture, the same methods can again be applied on the entire architecture with a different starting point based on the new requirements which suggested this new module.

Design changes within individual modules are not described either, but since the modules are ideally independent and highly integrated units, these design changes can be carried out as traditional product development projects. Possible implications however have not been described in the literature.

In the case of needed changes to an existing architecture or sub-architecture an evaluation of the new as well as the old layout for comparison is likely beneficial for more complex products. Many authors do give different methods for evaluating the technical quality of the platforms or architectures [Erixon 1998, Lyly-Yrjänäinen et al. 2005, Otto et al. 2007], while others present different metrics, in the form of checklists, for evaluating implementation and use of existing platforms [Riitahuhta et al. 1998, Kristjansson 2005]

Another kind of project is mentioned by [Fisher et al. 1999], who argues that cleanups in the product components and modules belonging to an existing product family will generate some beneficial effects (although not as many as if there had been no waste).

The typical platform projects described in various papers and dissertations from my own research group [Harlou 2006, Pedersen 2009, Hvam et al. 2008, Mortensen et al. 2008] are similar to the technical contributions described in the above, but the primary tool used for architecture building is the PFMP (which was briefly described in section 2.3.3). The applied approach which goes along with the PFMP is also a redesign of a product family, although [Kvist 2009] extends the approach into something more market-based. An important difference between architecture diagrams and the PFMP is that it is relatively easy to show module and component variety in the PFMP, which means that it can be used to showcase a messy and unorganized product family. The PFMP project is therefore more likely to start of as a cleanup-project, whereas the architecture diagram approach starts off from scratch or with a very ideal picture of the product family.

5.2. Platform difference

One reason for the differences between platform projects is the differences between the platforms themselves. Platforms are based on different *things* and these *things* are likely to influence the project.

5.2.1. Platform basis

Platforms are all based on sharing, but a number of different things can be shared. Depending on what is shared, product developers can have everything from no degrees of freedom to a high degree of freedom in design of the derived products based on the platform.

No degrees of freedom occur when the platform is based on a common modular architecture with a finite number of pre-designed units matching every generic placeholder module. Derived products are made by configuring the modular architecture according to specifications. Even though there is a finite number of pre-designed units the number of possible end-product may be very near infinite, when modules are combined. I call this a *closed common architecture* (Figure 71) and the platform is a skeletal platform. The Grundfos test bed platform described in section 4.2.2 is based on such an architecture. Many literature cases seems to describe platforms based on closed common architecture, because they base the design of the modular architecture on known product variants (see section 2.1.3).

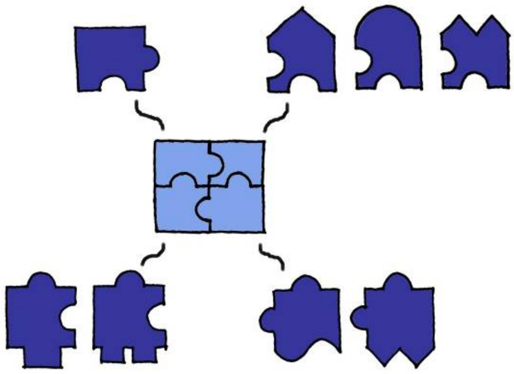


Figure 71: A closed common architecture is an architecture, for which there are a finite number of pre-designed units (dark blue) for each generic placeholder module (light blue).

A few more degrees of freedom are given, when the number of pre-designed units is not finite. Many products have architectures, where additional design units matching the requirements of the generic placeholder modules can be designed, when needed. I call such architectures *open common architectures* (Figure 72) and the platform is again a skeletal platform. The LEGO Bionicle figures and the LEGO mini-figures are examples of platforms based on such architectures. They each have a common architecture, which defines generic placeholder modules (e.g. an arm or a head) and their respective correlation and interfaces. Danfoss valves [Kvist 2009, Pedersen 2009] are also based on such architectures, and so are the LEGO LVM molds described in section 4.1.3. On a high level many consumer products like computers, cars, bicycles, television sets and phones each have common architectures (or maybe different kinds can be identified, which do have common architectures), it is however not safe to assume that the producers are utilizing this architecture commonality as a platform or are benefiting from this apparent similarity in any other way.

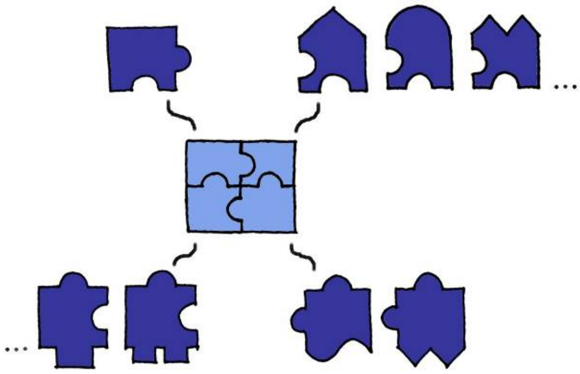


Figure 72: An open common architecture is an architecture, for which there are an infinite (symbolized with three dots) number of pre-designed units (dark blue) for some generic placeholder modules (light blue).

A subgroup of the *open common architecture* group is the *parametric common architecture* (Figure 73) group, where some generic placeholder modules are matched by design units, which only differ in a set number of parameters with an infinite number of steps. The Eagle Light machine on AKER MH oil rigs [Pedersen 2009] is based on such a platform. Parametric architectures are sometimes referred to as scale-based in the literature [Simpson 2004] although I include parameters, which are not size-related.

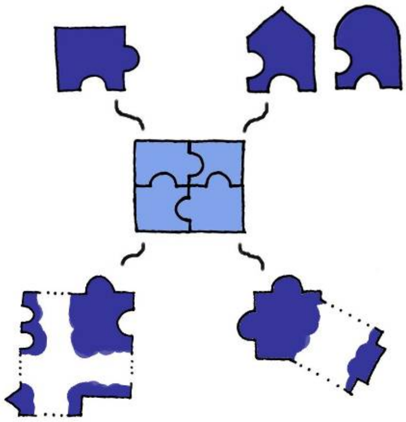


Figure 73: A parametric common architecture is an architecture, for which some pre-designed units (dark blue) are scalable.

An entirely other kind of platform is based on a common set of building blocks (the building block platform from section 2.1.1 and Figure 74). This kind of platform is not based on a common architecture. LEGO products are based on a platform like this. They share no common architecture (i.e. the architectures of a LEGO pirate ship and a LEGO fire truck are very different) but they are all built from the same set of standardized modules. This platform offers a lot of freedom, but is still limited by the fact that each module must comply with certain rules to be a building block. I assume that this kind of platform can be open as well as closed like those sharing architectures, but I know of no closed building block examples (i.e. where there is a finite number of building blocks available).

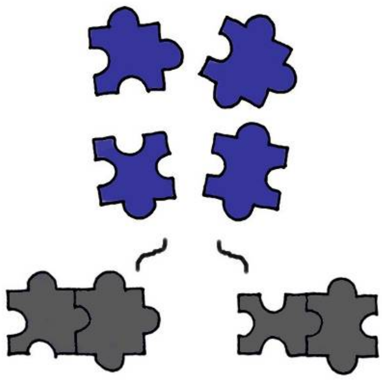


Figure 74: A building block system. The gray brick-combinations are products derived from the platform.

Finally some platforms are based on, but not limited to common building blocks (the cornerstone platform from section 2.1.2 and Figure 75), meaning that standardized modules may be utilized alongside special fully customizable units for building these products. The Grundfos pump-house molds described in section 4.2.1 are based on such a platform and so are some of the B&O products described in [Harlou 2006].

Cornerstone platforms resemble open common architectures, because both can be extended by an infinite number of design units. The open common architecture however imposes a set of rules on these design units (a specification goes along with each generic placeholder module), whereas there are no restrictions for the design units added to a cornerstone platform (excepting the fact that their interfaces must match).

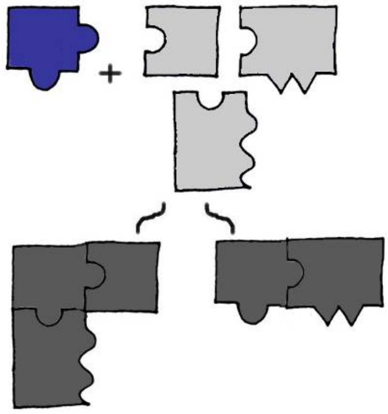


Figure 75: A cornerstone platform. The dark gray brick-combinations are products created by adding custom design units (light gray) to the platform.

5.2.2. Non-product platforms

The case examples listed in the above sub-section are mostly based on direct sharing between products. Most of the product families share either components or architectures. Most are based on product design.

There is however no apparent reason why platforms must be based on such direct sharing based on product design. Later stages in the company value chain could also be used as basis for platforms. The sharing of production equipment or the sharing of distribution channels could very well form the basis of platforms, which we might call production platforms and distribution platforms. The essential difference between these platforms and the product platforms described in the previous sub-section is that these other kinds of platforms would be based on sharing in other departments.

A production platform is a platform, which is about sharing of production components or architectures for a product family. If coordinated successfully and aligned with the products, this sharing should lead to various benefits like the product design platforms. Similarly such a platform would require various things of the products in order to comply with the platform. These requirements could be limits in sizes, shaping of components, material selections, and many other things based on the capabilities of the production platform.

A distribution platform works in the same way. Distribution platform offer benefits if certain requirements are fulfilled based on the capabilities of the distribution platform. Other types of platforms could also be designed based on other functions or departments in the value chain.

The necessary requirement for designing such platforms naturally is DFX (abbreviation of *design for x*, which is a generic term for *design for manufacture*, *design for assembly*, and others) [Fabricius 1994], or more correctly XFD (abbreviation of *x for design*, which is the opposite of DFX). When designing a new production or distribution platform it should of course be done with the products in mind, as depicted in Figure 76. How do we design a new production system or distribution system, so that current and future products are taken into account? Which limits do we create for the product designers?

Product platforms on the other hand should of course be done with the later functions and stages in mind. The product family should be adjusted to fit the production system and restrictions should be imposed on the designers that mirror the restrictions of the production system. This has already been described as seeking alignment and preventing harmful dispositions in section 2.6. The primary concern remains however how to create the variety we want to bring to the market, sometimes referred to as DFV, design for variety [Martin 1999, Martin et al. 2002]. This is shown in Figure 77.

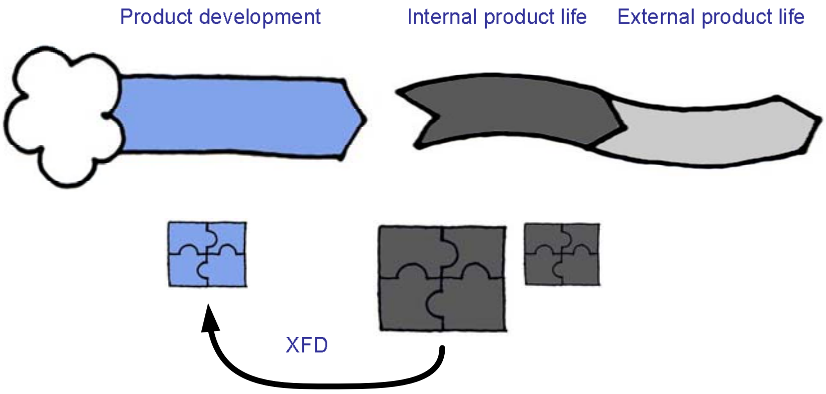


Figure 76: When designing a new platform for one or more stages in the company-internal product life (e.g. production, assembly) the primary concern is the alignment with corresponding platforms in product development. This is basically XFD for platforms – the system limitations and capabilities must be designed so that it matches the expected output of the product development process.

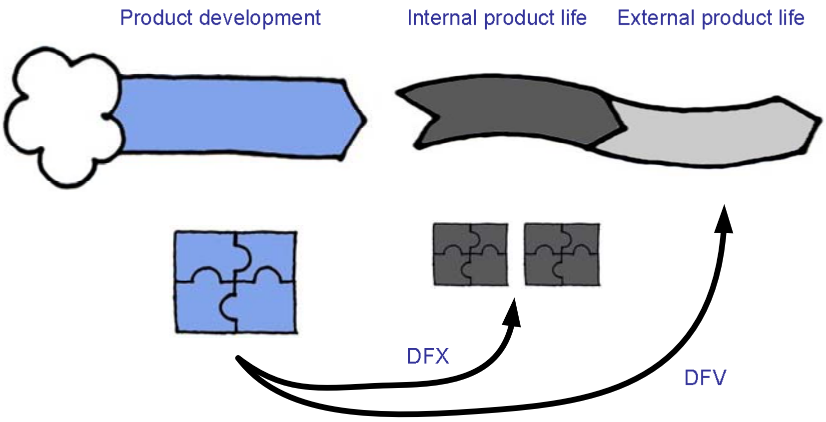


Figure 77: When designing a new platform in product development the primary concern is to design the platform to match the variety we want to bring to the market – DFV, while ensuring alignment with corresponding platforms in later stages of the product life – DFX for platforms.

The focus of this thesis is on product platforms, but some examples of production platforms were given in the case projects. The Low Volume project and the Module Mold project described in 4.1.3 and the Grundfos Molding Equipment project described in 4.2.1 can actually be defined as a production platforms, because in each case a general analysis was made of the product family, which should be supported (i.e. the assortment of low volume bricks, the total assortment of bricks, and the assortment of molded components including pump-houses, respectively). Similarly the Grundfos Test Bed project could be defined as a testing platform, but since the project was more about standardizing the existing assortment of production equipment than it was about creating a new platform based on the pumps, which should be tested, it resembles product platforms more than the molding equipment platforms do.

5.3. Kinds of platform projects

In the previous section, I gave examples of different platforms, which of course influence the platform projects and make them different. It is clearly one thing to build a common architecture, and something else to build a set of common building blocks. Another thing, which influences the platform projects, is the starting point and the aims of the project.

As written in the beginning of this section platform projects, which aim at introducing new platforms, have been heavily discussed in the literature.

The case projects described in section 4 however, are clearly not all this kind of platform project. First of all, almost all of the projects add to or modify an existing platform. Almost all of the projects also deal with only a subsection of a bigger platform.

These subsections of a bigger platform can however always be seen as one or more confined modules, and because they are confined they can be modified independently and treated as products in themselves. There is therefore no practical reason why a subsection of a platform should be treated differently than a complete platform. Both should comply with all interfaces related to the surrounding systems.

It does however matter if the platform is made from scratch or modified from an existing design.

5.3.1. Platform novelty projects

I use the term platform novelty projects for projects, which aim to create a new modular product family from scratch. There might have been a previous product family or a product range covering the same marked area or the same function, but none of these products have modules which will be reused in the new product family, and the old products will all be obsolete.

The term is also applicable for a new family of modules, which can be added to an existing platform without modifying this platform significantly. Again this family of modules might replace an older set of modules, but nothing of significance is reused (except for the interface to the remaining platform), and all old modules will be obsolete.

The Low Volume Mold project, the Power Functions project, and the Panels project at LEGO (see section 4.1.3) are examples of such platform novelty projects. The Power Functions and Panels brick families completely replaced older bricks, which were made obsolete during the project. The Low Volume Molds did not replace any exact group of existing molds although a large group of molds were later out-competed by these new molds.

Common for these projects have been that they start with a heavy analysis of the product or module need.

5.3.2. Platform redesign projects

I use the term platform redesign projects for projects, which aim to change an existing modular product family or family of modules. In this case it has been realized that the existing assortment is too expensive to replace completely and/or that some parts of it are reusable and of a high quality.

The Module Mold project, the Wheels project, and the Walls & Windows project at LEGO (see section 4.1.3) are examples of such platform redesign projects. Likewise the Grundfos Molding Equipment project and the Grundfos Test Bed project were such projects.

Common for these projects have been that the aim was to create a new and more modular architecture and product family heavily based on existing concepts and modules or components. The Module Mold project modified the Low Volume platform and expanded it. The Wheels project and the Walls & Windows project created a more modular sub-assortment based on existing as well as new components, and the Grundfos projects build on existing concepts and reused components as well as architectural elements in new common modular architectures.

This also means that a common element for these projects is a heavy analysis of the existing product or module assortment.

In my research group we are most familiar with this kind of project, because in most of our known case examples the companies cannot afford to start from scratch and good concepts within the existing assortment could be expanded into the remaining assortment. The product family master plan (PFMP) tool [Harlou 2006, Kvist 2009, Hvam et al. 2008] is dedicated to this type of project, where the platform project starts off with an analysis of the existing product assortment. This tool has consequently been applied in many cases, including the Grundfos cases described in this thesis.

5.3.3. Permanent clean-up projects

Creating a good modular structure or a set of good common modules is of course essential for platform-based product development, and this part is also the most documented part as described in the beginning of this section. The LEGO platform story however shows that having a good modular structure is not enough.

To ensure that the platform performs as intended it is necessary to design the behavior of the platform users. How should this platform be used and updated? Which new modules or components should be added to the platform and which old modules or components should be removed from the platform?

The Wheels project, the Minifigure Accessories project, the Walls & Windows project, and the Shapes project were all projects, where one of the key aims was to clean-up in the assortment and make sure that this clean-up was permanent. The Wheels project and the Walls & Windows project succeeded in doing this by restricting the addition of new bricks heavily. Very few Wheel components or Walls & Windows components will be created in the future. On the contrary, the Minifigure Accessories and the Shapes project would have to allow a lot of future bricks to be added to the platforms, but these projects have ultimately not been very successful.

The conclusion here will however not be that the only way to ensure permanent order in the module assortment is to heavily limit the addition of new modules. The Walls & Windows platform does put restrictions on new bricks, but there is still an infinite number of possibilities for new bricks within the platform. And the reason that the Minifigure Accessories an the Shapes project failed to ensure permanent order has to do with the fact that there is no accepted procedure for evaluating the difference in Minifigure Accessories or Shape components (the review sheets for UGS-classification in section 4.1.2 generally result in special for all Minifigure Accessories components and universal for all Shape components), making it very hard to differentiate the individual bricks and decide which to keep and which to remove.

In more complex cases the PFMP tool could also be used for clean-ups (it will not ensure the permanency though), it has however not been utilized in the described LEGO projects because LEGO products have no common architecture (which is a requirement for the PFMP – see section 2.3). I will return to how to deal with permanent clean-ups in section 7.1.2.

Ideally all platform projects should include elements of a permanent clean-up project. This part is however seldom mentioned in the literature. In LEGO however, emphasis has been put on this kind of projects, because it has been realized how much money LEGO is spending every year on disorder in component assortments.

5.3.4. Modular integration projects

Sometimes it pays off to integrate instead of modularizing. Even in modular products it may still be wise to sometimes integrate several modules, rather than always increasing the modularity.

The Frames project in LEGO is an example of such a project. Smaller existing components can actually be assembled into something which is roughly equivalent to the Frames components, but integrated Frame components are less expensive, more robust, and easier to handle for the customer.

5.3.5. Modular quality improvement projects

As has been stated before, the modules in a modular architecture are well-confined, which makes it possible to run independent projects affecting only some parts of the modular architecture. A typical project of this kind is a quality improvement project, where one module or a set of modules corresponding to a generic placeholder module is replaced by better versions.

The Ball Joint project at LEGO is an example of such a project. The project itself did not increase, decrease, or change the modularity or variety of LEGO products in any way (and it can therefore be argued if it was really a platform project), but because of the already existing modularity, the improvement of the ball-bearing concept affected many different components, which copied this concept.

5.4. Relating platform projects to other project types

Platform projects are different from product development projects in many aspects. This is primarily due to fact that product development projects end with a product, which is then handed over to the production. The product life does not end, but the development does. Development of a product can be seen as a process of preparing for the remaining product life.

Development of product platforms is similarly a process of preparing for the product development of a number of derived products, but this process often has no end, since most platforms are intended to be updated along with the technological evolution (even though this has not been focused much on in state-of-the-art literature). For managerial purposes it is however often convenient to split up the continuous platform development process into smaller projects, which do end. Examples of such projects are given in the case material (section 4) and generalized in section 5.3. It is however important to note that these projects do not truly end, but are in reality sequential tasks in the overall continuous platform development project and should be treated as such.

In this sub-section I will compare platform development projects to product development projects – not because they are very similar, but because they are related and because they are often carried out by the same employees, the product developers.

First however I will compare platform projects with industrial technology/research projects, which are more similar to platform development projects though not identical.

5.4.1. Technology/research projects

Many companies, which develop new products, have a Research & Development department. As the name itself suggests there are two typical activities going on in such a department: Research projects and Development projects (and maybe a number of projects which can be classified as in-between these categories).

Typically these two projects are differentiated in the following way:

Industrial product *development* projects are characterized as having a certain output, a new product. The process has a relatively high likelihood of success, and the schedule can be relatively fixed. The consequences of failure can however be dire – both in terms of spent investments and failed market strategy.

Industrial product *research* projects on the other hand are characterized as having a fuzzy output. The process has a relatively high risk of failure (i.e. of showing no or limited results), and the schedule is highly uncertain. The consequences of failure are however not big, since technological evolutions are rarely so anticipated, that they are actually counted on and invested heavily in.

Like platform development, product research can be continuous – continuously preparing for and making things available for the product development projects. In this aspect platform development projects are very similar to research projects.

Sometimes a new technology or concept derived from a research project can be utilized as a product platform, where a number of product variants can be created based on the same core technology.

The obvious disadvantage of using the result of research projects as product platforms is that they are often unproven and risky, which is not very beneficial for platforms. Additionally, new technologies will often eventually be outdated themselves, and a platform building on such a technology will therefore be born with an expiry date. Technology-neutral platforms on the other hand have much better chances of remaining relevant.

Many of the original contributors in platform-based product development were based on case studies in the PC software industry [Meyer et al. 1997, Baldwin et al. 1997], and therefore the concept of product platforms have been slightly confused with the concept of PC software operating systems such as Microsoft Windows, Apple, and Linux which are sometimes called software platforms.

A more general confusion also exists between technology projects and platform projects. Some companies and authors [Meyer 1997] define their core technologies as platforms [Elgård 1998], either based on the fact that they can derive a number of product variants from one core technology, or based on the terminology from software engineering.

Following the general definitions and perceptions in section 2, there is no reason why PC operating systems or core technologies cannot also be product platforms, since they resemble the cornerstone platforms described in section 2.1.2. It is a fundamental concept of PC software programs that they rely on common operating systems, which they build upon. PC software products are sold as extensions to the operating systems, and do not work on their own. They can therefore be considered add-on modules to the PC operating system platform.

PC operating systems are however atypical kinds of platforms in the way that they are often not very static. They are frequently updated and patched to accommodate for new hardware products and flaws in a way, which is only possible for software products. More importantly companies often strive to continuously improve their operating system or core technology, because it is necessary for increasing the product performance. This very much contradicts the traditional perception of what a cornerstone platform is. In my perspective the true cornerstone platform is made up of those core modules within the operating system, which makes these frequent updates possible, and which remain unchanged by the updates.

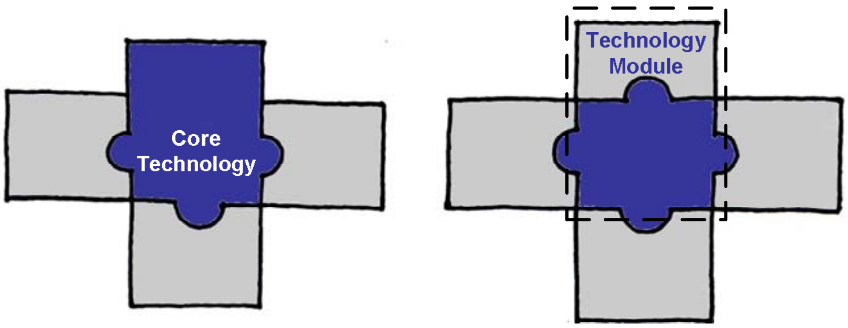


Figure 78: Technology platforms. Left: Core technologies are often employed like platforms, but if the technology evolves, this is not very beneficial. Left: Technology should instead be confined in replaceable modules (from an outside perspective the modularization might not be visible).

The other parts of the PC operating systems can be perceived as technology modules. They are continuously modified or replaced due to technological advances. Including these in the platform will only make it less static (Figure 78).

In section 3.1, I stated that the continuous evolution of technology and market lead to a continuous evolution of the platform. This is not contradicting the statements in this section.

Continuous platform development means continuous extensions, additions, replacements and modifications within a product architecture – especially but not exclusively within the generic placeholder modules. Whether the company operates with a common product architecture as a platform or with a set of common modules (i.e. a cornerstone or building-block platform), the platform core itself is not going to change significantly.

5.4.2. Product development projects

As stated in the beginning of this sub-section, product development projects and platform development projects are two different things.

They are similar because they can both be perceived as processes of preparation. Product development is a preparation for the remaining product life cycle. Platform development is a preparation for coordinated product development. It is possible to produce a product without going through a product development process, although most professional companies do product development. Likewise it is also possible to develop many products without creating a common platform, and most professional companies are trying to do platform development (or some kind of coordinated product development at least).

Platform development is however about creating generic things and things which can be shared, while product development is about creating dedicated things. The tasks of designing the individual modules needed for platform-based product development (common standardized modules or design units matching generic placeholder modules) can however be handled as independent traditional product development projects, where all special requirements for interfaces or functions derived from the platform is included in the product development specification.

5.4.3. End-process vs. Continuous

Another important difference between product development projects and platform development projects mentioned in the beginning of this section is that product development projects are end-processes while platform projects typically are supposed to be continuous (Figure 79).

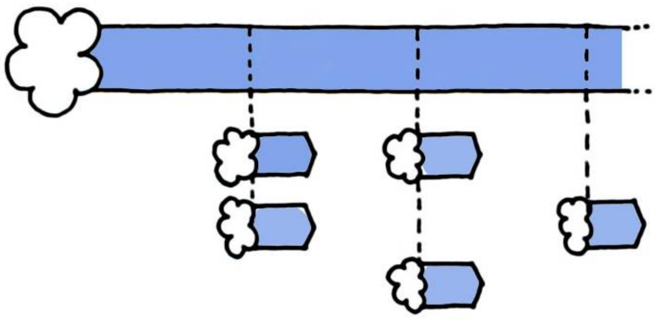


Figure 79: Platform-based product development. The platform development process (top arrow) is continuous, while the product development processes (lower arrows) are end-processes.

Companies trying to manage platform development projects as end-processes with a deadline, where resources consequently are removed from the platform project after the deadline, will find that the long-term benefits and workflow implementation will diminish.

This was observed in the Grundfos Molding Equipment platform project, where the platform was not extended with any new modules after the project was closed, even though this was planned.

Similarly when the LEGO Minifigure Accessories platform project was closed later components were not added to the overview the project had produced and consequently only effects (in terms of component assortment clean-up) realized within the project time frame was gained.

A number of other cases documented by [Munk 2009] confirm that platforms, which are developed in end-process projects, generally stop evolving. Some companies succeed in using the platforms afterwards in product development, but the platforms are not extended or improved as a part of the development tasks afterwards.

In cases where the platform development is continuous, this is obviously not the case.

The Grundfos Test Bed platform project assigned a number of employees to permanently improve and develop the platform as a major part of their work responsibilities.

Also a number of LEGO platform projects, which are heavily anchored in the DesignLab organizational unit, are still being developed upon. This primarily includes the original building system platforms LEGO System, LEGO Technic, and LEGO Duplo, but newer initiatives such as the Universal Elements platform project and the Screening projects as well as a number of undocumented minor projects are also constantly being evaluated and revised.

5.4.4. Benefits, costs, and risks

A major difference between platform development projects and product development projects are the benefits, the costs, and the risks of the project.

The cost-benefit purpose of a product development project is basically to launch one or more successful products in the marketplace, which will eventually make a high enough turnover for the company so that the accumulated investments and costs are paid for and so that a reasonable return-on-investments is met. Additionally benefits of product development can be acquired knowledge, training of employees, company promotion, product family promotion, improved competitive position, etc.

The cost-benefit purpose of a platform development has already been described in section 2.6, but it is basically to improve the economies of scale and/or reduce the order lead-time so much that it outweighs the investments and costs of introducing and maintaining the platform. Additional benefits of platform development can be improvement of product quality (e.g. Ball Joint project), training of new employees (e.g. Test Bed project), increase of the product range, etc.

Risks are similar in kind and reason, but vary in size between the two kinds of projects.

Risks of meeting cost estimations or deadlines will often be higher for platform development projects, simply because the company has less experience in carrying out these kinds of projects. The consequences of failing deadlines should however be smaller, since platform development rarely leads directly to product development. In these aspects platform development projects resemble research projects.

Technological or technical risks tied to decisions on product concepts and functional and physical architecture should be less because platforms should primarily use well-known and tried product concepts and solutions. For larger product families, technical risks could however increase due to the complexity and untried module combination possibilities in the total product family.

The consequences of technical failures can however be dire in platform development. A fundamental flaw in a core part of the platform could ruin a complete product family – and not only one product.

When the risks of the platform development project has been taken and hopefully successfully avoided, there is significantly reduced level of risk in the product development projects afterwards.

Product development projects based on platforms should experience a significantly reduced level of risks in cost estimation, scheduling, and technical flaws – simply because the size of individual projects and the number of uncertainties are reduced.

5.4.5. Decision-making

Decisions in product development projects are important because they form dispositions for the remaining product life cycle [Olesen 1992] (Figure 80). Bad decisions are likely to impede the product permanently. The worst thing about dispositions is that they are often hidden, and the product can therefore be unwittingly permanently impeded.

Decisions in platform development projects are even more important because they form dispositions for our future product development projects (Figure 81). Bad decisions here are likely to impede product development – and therefore *all* of the future products. Companies operating with less formal product platforms effectually hide the dispositions of their platforms, and *all* of their products can therefore be unwittingly impeded.

Employees making decisions in product development use their technical knowledge and understanding of the business to affect the benefits and costs of the product development project mentioned in the previous subsection.

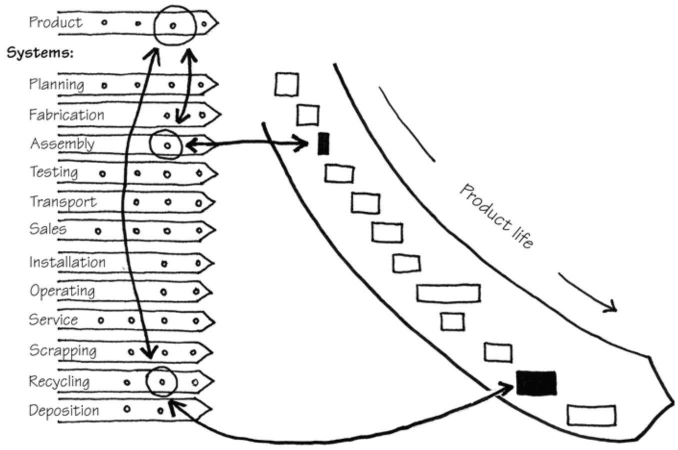


Figure 80: Product development dispositions [Olesen 1992]. Decisions in product development must take into account the effects the product will create in the later stages of the product life.

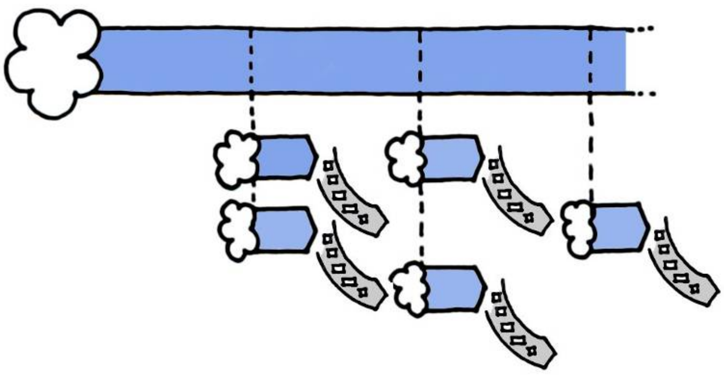


Figure 81: Platform development dispositions. Decisions in platform development must take into account the effects the platform will create in development of derivative product variants and in the later stages of the life of these product variants.

Employees making decisions in platform development use the same skills to affect the benefits and costs of the platform development project, but since these benefits and costs are radically different, the decisions themselves are different. Therefore making such decisions requires at least an understanding of the benefits and costs of platform development.

Additionally, the cross-company scope of the benefits of platform-based product development also requires that, those making the decisions have a broad product responsibility.

At LEGO product platform responsibility has been given to a few dedicated business units, where previously-mentioned DesignLab is the most notable. Since this business unit is initiating many new platform projects continuously, it has built up a level of expertise, which makes more capable of running these projects than other product developers in LEGO. Also since DesignLab is not involved directly in any product novelty projects, it can remain unbiased and take a broad responsibility of the component assortment.

Similarly Grundfos TC is employing a platform architect, who will participate in all of the platform projects at Grundfos TC. This person will (initially assisted by experienced consultants) ensure the platform expertise in the various future platform projects.

5.4.6. Documentation

Many companies have requirements for the documentation of product development projects. The most important part of this documentation is often the 2D assembly drawings of the product and its components, which will be used for procurement, production, and assembly. A number of additional drawings and bill-of-materials and similar lists or structures may also be produced. Finally the working documents of the project can be saved and used for future reference, but this is rarely done [Wallace et al. 2003].

For platform projects documentation takes a whole new meaning. The primary documents are likely to be different views and charts, which show different aspects of a common architecture or building principle. This may be in the form of structural block-diagrams, functional block-diagrams or case-examples. The first major difference is that there is no equivalent to the 2D drawings of product development projects (although a number of similar diagrams have been proposed by [Hölttä et al. 2003] – these have another purpose though), nor is there an equivalent for the bill-of-materials although the PFMP [Kvist 2009] – see section 2.2 – can be used as such.

The second major difference is the working documents. Because there is no equivalent to the 2D drawings of traditional product development, there is nowhere where the results of the project are shown. The primary documents will only show some aspects of the project, and therefore a larger set of documents are needed for the platform to be used. Also, since platform projects are not end-process projects, the working documents are never really finalized, but are continuously needed to pass on the information to new employees, who take over the platform development project.

At LEGO a number of wall-sized charts are explaining the building systems and their rules. Since there is no common architecture, it is instead all of the common components that are shown. This library of components is constantly updated, and new subgroups are constantly revised and described.

Grundfos TC similarly has a library of mold inserts (i.e. the collapsible core variants), which are available in a dedicated and easily-accessible place in the databases. The Test Bed project resulted in a common architecture, which is described in a PFMP and a generic 3D model, which is built up of generic placeholder modules.

6. Finding and Screening Platform Projects

This section aims at answering the research questions stated in section 3.2 or confirming the hypotheses from the same section.

How do we find platform project ideas?

and

How do we prioritize platform project ideas?

The state-of-the-art literature is not very helpful in answering these questions, since most works are about designing or improving the architecture of a given product family or about the effects of platforms in general.

A few authors have however briefly touched the subject:

[Halman et al. 2003] states that: “Most platform decisions are not concerned primarily about whether to invest in a platform or not, but instead about the valuation and strategic selection among platform alternatives.”

[Otto et al. 2007] states that: “Ideally, a simplified analysis would be available at the very early concept phase to evaluate platform alternatives.”

Both authors recognize that there may exist platform alternatives and that these need to be evaluated before implementation.

No authors specifically details from where or how these platform alternatives originate or provides a simplified analysis for this evaluation.

Based on the case studies described in section 4, I have however gained some insight into this nascent area of research, which I will present in the following.

6.1. The origin of platform projects

When a new platform project idea has to be found, we have to know where to look and also when. I have already covered what to look for in section 5, but in short we are looking for platform novelty, platform redesign, permanent cleanup, modular integration, or modular quality projects and these may be based in product development or any later company-internal product life phase.

6.1.1. Introduction

Many of the platform projects referred to by my research group (see section 4.3) have not been initiated with a thorough search for alternative platform projects. Most have instead begun with an introduction speech on platforms in general with case examples, which immediately have lead the company employees to suggest similar projects based on recognized similar situations or aims. Many platform projects have spawned from these intuitive ideas, which are often obvious platform projects, which we often refer to as *low-hanging fruits*.

After an extended period of cooperation with a company it seems however, that these low-hanging fruits become exhausted and more effort must be spent to find good platform projects.

This has lead to the conclusion that the lack of expertise in platform-based product development (or lack of confidence in this expertise) often prevents companies from initiating even very obvious projects even though they themselves recognize the potential.

Usually using low-hanging fruits is preferable, since the company employees are already aware of the problem and are therefore more susceptible to solutions related to the same problem.

The following findings and search advice describe the circumstances under which low-hanging fruit appear and can be additionally applied when the first low-hanging fruits have been exhausted, as is the case at LEGO Company.

6.1.2. Timing

Good platforms are supposed to be permanently relevant, but initiating a platform project requires the correct timing. As with most statements about platform-based product development this can not be ultimately confirmed, some cases however indicate the validity of the statement very much.

The *Wheels* project (section 4.1.3) was unsuccessfully initiated several times before it finally came through. Several things were changed in the project between each try, but in the larger perspective it is much more likely that it was outer circumstances which finally changed failure to success.

From the different cases it appears that there are two things to consider about the timing of a platform project. One is the company maturity and the other is the market maturity.

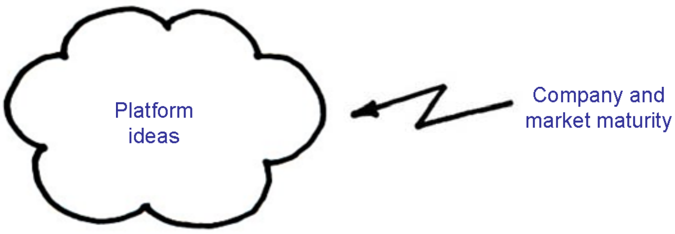


Figure 82: Platform ideas require correct timing, meaning that the company and market need to be mature for them to succeed.

Company maturity

The company maturity is a measure of how ready the company is to initiate and implement platform projects. Since platform projects are long-term investments, companies must be sufficiently confident that they will ultimately be able to leverage the benefits.

The burning platform situation at LEGO following the economical results of 2003 and 2004 (see section 4.1.2) directly lead to a policy, which LEGO employees themselves referred to as “there are no sacred cows”, meaning that they committed themselves to being ready for change and that all existing procedures should be revised.

The success of the *Wheels* project and the following shift in responsibility for development of universal components from product designers to DesignLab should be seen in this perspective. This was also one of the conclusions in a series of interviews with product designers at LEGO [Harder et al. 2008].

Although not as apparent as in LEGO, a similar burning platform situation exists at Grundfos TC. The threat of external competition and unhappiness of the customers based on delivery-time and price has lead to a situation where several departments in Grundfos TC must improve their general performance or face outsourcing.

In the absence of a true burning platform situation some companies have succeeded in creating a future one. Danish-based valve-producer Danfoss and Norwegian-based oil rig-producer Aker Solutions are very successful these years and make a decent profit, both state however that if product development performance does not improve radically in the coming years they will have a burning platform.

In extension of this conclusion a number of possible consultancy and research projects planned by my research group, which never were properly begun (and therefore never documented), could be explained by a lack of burning platforms in the cooperating companies.

Another finding is that a burning platform is often not enough. Management must have a certain interest in platform projects; because the projects require their commitment to be successful (see section 2.6.5).

LEGO has really struggled with this problem, because management traditionally has perceived platform activities as being technical in nature and therefore strictly up to the individual product designers. Gradually this perception has changed though as component frames and classifications has been top-down implemented in the organization. Still, because many platform benefits are not directly quantifiable and has very much to do with product strategy, management needs to involve itself deeper than on the financial (i.e. quantifiable) level.

Our strategy of linking platform projects directly with company strategy documents (i.e. the *Shared Vision* in section 4.1.3) has successfully secured management interest in later platform projects at LEGO. Naturally this requires such documents to be available.

Market maturity

Market maturity is really about how ready the product or technology and the market are for a platform. To be able to create a relatively static platform it is necessary to have a stable product concept, meaning that the product must maintain is overall architecture or constitution for a reasonable amount of time. This is often not the case in the earlier phases of a market life-cycle for a product family.

[Martin 1999] is showing an s-curve with three distinct phases, Figure 83. The traditional product platform which emphasizes economies of scale and such effects is mostly directed at the latest phase, where the product is stable and mature and the market is well understood.

Another kind of platform is needed in the middle phase. Here it is necessary to create rapid changes and improve and optimize features. This requires continuous development of platform components.

The first phase in Figure 83 is not ideally suited for platform-based product development. As described in section 2.4.2 this is where innovation strategies are more appropriate.

The same division is shown in [Elgård 1998], where the era of incremental change is the platform strategy era and the era of ferment is the innovation era – see Figure 84.

In some cases incremental changes are so rare or cost optimization so tough in the end of the final phase, that a modular architecture is not rational anymore and integral concepts should replace modular ones.

All of the cases, which this thesis builds upon, are dealing with relatively mature product families and I can therefore not present any cases, which proves that platforms for immature products fail. The fact that there are no cases combined with the lack of literature claiming that such platforms are possible indicate however that it is very likely that platforms require market maturity.

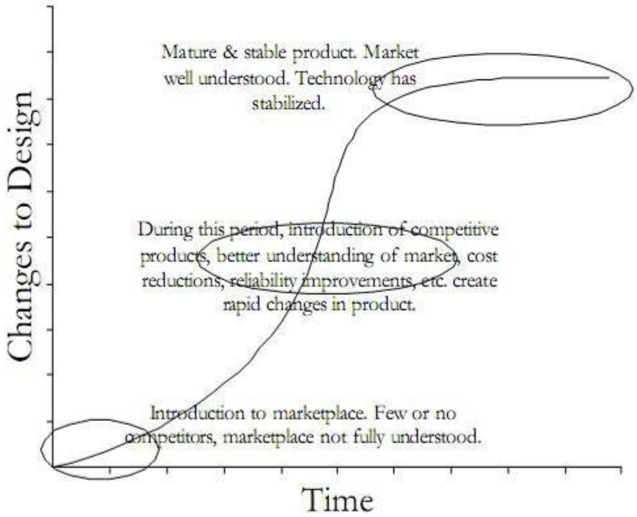


Figure 83: Market life-cycle of a new product category [Martin 1999]

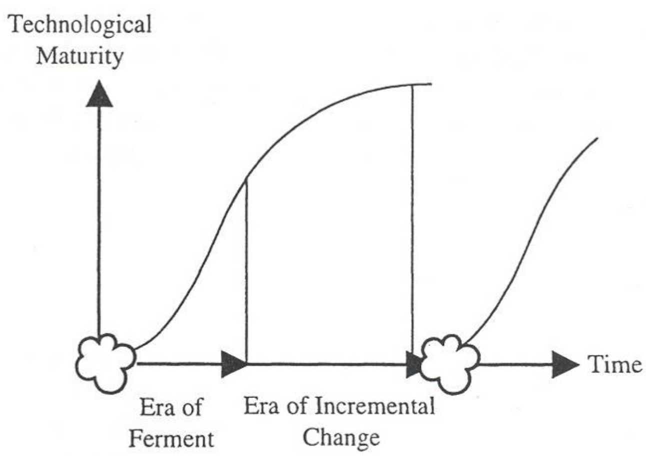


Figure 84: Technology life cycles [Elgård 1998] derived from [Sanderson et al. 1997]

In some literature case studies (e.g. [Sanderson et al. 1995]) it is not clear how mature the products are before the platform is designed, it is however clear in the same case studies that the fundamental architecture or constitution of the product remains the same when platforms are applied, so this does not conflict with my statement of market maturity.

6.1.3. Place of origin

Product platform ideas can originate from several different parts of a company, but different kinds of platforms are suggested from the different parts of the company.

Marketing and sales

Marketing and sales persons typically perceive products as different configurations of features, functions, or specifications. This perception is very useful in the early stages of platform development, since a product architecture can be defined based on these different configurations. Furthermore marketing and sales persons sometimes lack the technical insight, which often hampers engineers in rethinking a product family, and instead have insight into which product features are valuable for the customer – an extremely important competence in a platform project. Because of this marketing and sales persons are highly valuable in the early stages of platform development and as idea generators for platform projects.

None of the project ideas described in this thesis originates from marketing or sales persons. This fact may be coincidental or it may be the consequence of a general lack of insight or interest in platform-based product development by marketing or sales persons. Generally the concept of product families as a common entity based on a shared brand name is recognized, but it is doubtful how much sharing in general is valued by marketing and sales persons. Based on random discussions with marketing and sales persons in various companies, I will state that the concept of commitment to already created platforms and existing platform components seems sometimes in conflict with the general performance indicators of these same persons.

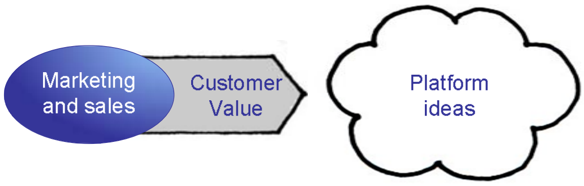


Figure 85: Persons from marketing and sales departments can suggest platform projects, such projects often have a high customer value - but may lack technical realization.

The argumentation that innovation and platform-based product development does not conflict, which was presented in section 2.4, may be useful in convincing persons who lack deep insight in platform-based product development of this non-existing conflict, and a presentation of the burning platform in the company (if one such exists) may convince key marketing people that platform project are needed and that they are required to participate.

Downstream departments

Employees from production and distribution typically suggest production platforms or other non-product platforms (see section 5.2.2). In a company where the products are optimized and cannot be improved significantly this is very useful. However, in most companies introducing a production platform (or other down stream-based platform) without changing the product platform is risky, since we risk building an optimized delivery system for a problematic, faulty, or expensive product family.

At LEGO there is an overall belief that the fundamental product building system is good and in many regards ideal, the studs and tubes and basic geometry of the LEGO bricks are not likely to change. Based on this fact it is possible to build production platforms, which are based on the LEGO bricks.

The LEGO Low Volume Mold project and the Module Mold project are examples of production platforms suggested by the production department at LEGO. Since both platforms are relatively broad and can be used to produce many kinds of LEGO bricks, they have never been very risky, but are based on rational facts and experience. If, however, the LEGO products were undergoing a significant change in composition these projects would have been highly risky, since they were primarily based on existing and past LEGO bricks.

The Grundfos Molding Equipment platform is more uncertain. It remains unclear if the pump-houses will continue to be produced with the current connections and this is perhaps the reason why other connections have not been implemented in the platform. It is simply too risky to design platform components based on connections, which will maybe never be used again. This is clearly an example of a case, where further development of the production platform requires further development of the corresponding product platform.

When employees from production or distribution do suggest product platforms, they will often do so based their insight in the existing production or distribution architectures or capabilities. These suggestions may open up for inexpensive variety possibilities or new product opportunities which could lead to genuine innovations in the products. However, there is a strong risk that these suggestions are not aligned with customer preferences or values, and they should therefore be thoroughly examined before being put into the product pipeline.

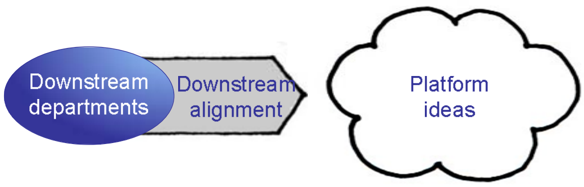


Figure 86: Persons from downstream departments can suggest platform projects, such projects are often aligned with downstream architectures - but may lack customer value.

At LEGO a newly founded special unit based in the production department suggests new product platforms based on new production capabilities and sub-supplier capabilities. These suggestions are innovative (they are not minor improvements) in the sense that they are new and open up for new product opportunities, but most do clearly not fit into the LEGO product portfolio. Still some of the ideas might fit (it remains uncertain presently) and if they do, they will make the whole effort worthwhile. Unfortunately I cannot go into more detail in the description of these suggested concepts due to confidentiality.

Other kinds of platform suggestions from production include product platforms based on a late differentiation strategy in production. Here the platform may be a common generic premature product like the generic collapsible core platform component in the Grundfos Molding Equipment case.

Inviting down-stream employees to participate in platform development is not only very useful, but actually it is a requirement since any good product platform must be aligned with the architectures and platforms of all the down-stream activities, as I have stated several times before.

In LEGO this has sometimes been a problem, since many down-stream employees have more pressing responsibilities than commenting on possible future product families on a relatively high level, and since employees in these functions are trained in analyzing and modifying existing equipment or products and untrained in conceptual thinking. Their commitment in platform development is therefore sometimes not full.

This is but one of the reasons why it may take time to implement platform-based product development in an organization. Luckily the continuous advertisement at LEGO by platform thinkers and for some part management are gradually making results and more employees are now prioritizing platform development, although with slightly different agendas and perceptions.

Product development

Product designers are the natural source of suggestions for product platform. Their technical insight in the products and their compositions makes them well equipped for suggesting new platforms, and often if they are participating in many development projects and influencing the design of many products they will naturally reuse functional units or subassemblies from one project to the next.

The primary personal reason for designers to apply reuse in product development is to save time and reduce risk. There is also a strong tendency that they only reuse components, which they themselves have designed and not those of their colleagues. This I have confirmed through talks with product designers both at LEGO and at Grundfos.

The Grundfos Molding Equipment project also highlighted that even though designers were reusing mold components in their CAD systems, this information was previously not passed on downstream, and thus lost. The CAM programmers always started from scratch even though much was unchanged between the designs. This in effect prevented all of the downstream benefits of sharing.

Based on these observations, I believe that even though product designers will sometimes suggest platform projects, they will often do so for their own benefit. This is of course often not a bad thing, since they should be encouraged to applying sharing and reuse to save time and reduce risk, there appears however to be a tendency for them to stop too short, and as a result the full potential of platform projects may not be realized.

The concept of *knowledge platforms* is related to the above. With knowledge platforms I refer to a phenomenon, which some companies advocate, namely the concept of sharing on an abstract level [Harlou 2006]. Some companies find it impossible to apply full sharing of actual components or architectures and instead allow a modest modification process to take place whenever a product is developed.

Such knowledge platforms are beneficial for product designers because they essentially offer a pre-designed template, which can be modified to fit a broad spectrum of needs. Again this can save time in product development and reduce risks, but the downstream benefits of platform-based product development cannot easily be leveraged, because these effects generally depend on things being exactly identical and because it can be hard to inform downstream employees on what they can actually depend on.

For those companies, which cannot apply full sharing, but allow modifications, I will instead recommend the approach in the Grundfos Molding Equipment project. The collapsible core platform is never presented as a shared abstract platform, which is then modified to match customer requests; it is instead a shared premature product platform, which is then modified to match customer requests. The difference is important, because the downstream activities can be based upon the latter, but not on the first.

Common product architectures (when they are not combined with standardized shared components) are also abstract phenomena, which are often suggested by product designers. They generally share the benefits and downsides described in the above to some extend. Common product architectures however do still standardize composition and interfaces though and this is especially useful for standardizing processes in product assembly.

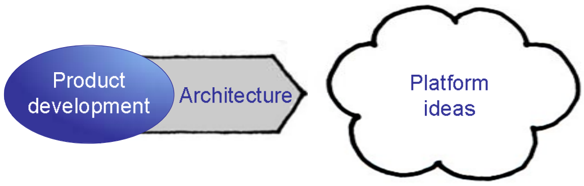


Figure 87: Persons from the product development department can suggest platform projects, such projects often features common architecture or compositions - but may lack downstream effects.

When all of this is said, product designers remain the most central persons in product platform projects and as idea generators for these projects. The comments in the above merely suggest that relying solely on product designers will not leverage the full potential of most platforms.

6.1.4. Person of origin

As platform development responsibility is often not placed in one specific department, it may some time also be meaningful to search for specific persons, which can participate and suggest new platform projects.

Insightful experienced workers

Insightful experienced persons, who have a deep technical understanding of the products and the manufacturing process, are essential in platform projects. These persons often have ideas for improvement or change of products or processes, which can be used in platform projects, but lack the ability or opportunity for proposing and getting their ideas approved in the organization. These insightful (often senior) employees are often the true source of the before-mentioned low-hanging fruits.

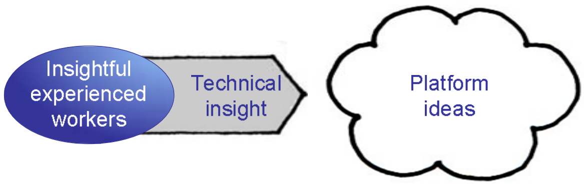


Figure 88: Insightful experienced workers can suggest platform projects, such projects are often technically sound and aligned - but may lack sufficient platform insight to present effects.

The Grundfos TC approach to platform development evolves around this concept. The strategy has been so far, that within every division the most insightful experienced worker (or workers) are identified and made to participate in a platform project aiming to renew the manufacturing process or products or the division. Both the Molding Equipment project and the Test Bed project have leveraged on the experiences and technical knowledge of these insightful employees and both successes have without a doubt required their participation.

The platform approach at LEGO also evolves around insightful experienced workers. The senior designers employed at DesignLab all have many years of experience and a vast knowledge of the LEGO bricks and the building systems. Traditionally their insight in production and distribution have however been reasonable but not quite adequate for many platform purposes, and that is perhaps one of the primary reasons for the strong product-focus and few downstream effects of many of the platform projects carried out in DesignLab.

Insightful experienced workers in the production, assembly, and packaging departments have supplied platform ideas for the Low Volume Mold project and other platform projects, which will be documented in [Munk 2009], but the frequent changes in production and distribution setup following the various sourcing and outsourcing strategies between 2005 and 2008 have made it difficult to find experienced workers. It has also been near-impossible to align product platforms to a great extend with the frequently changing production and distribution architectures themselves.

In conclusion the platform projects at LEGO must simply remain product-focused and disregard downstream architectures until an overall layout of these architectures has stabilized, but when this happens, employees with technical insight in this new setup are required in the platform projects.

Managers

Managers often have responsibility of several products, several production lines, or similar things, and because their responsibility is not restricted to any single product, product line, or similar thing, they often are more focused on commonality and variety between these things than their employees.

Managers also frequently have more strategic insight and a greater understanding of where the company is going in the longer perspective.

Figure 89: Managers can suggest platform projects, such projects are often strategically aligned - but may lack technical realization.

Because of these traits, managers are very useful in platform projects and for suggesting platform projects. They actually supplement the insightful experienced worker nicely, but rarely take active part in platform development projects.

Additionally, managers are of course interesting for platform projects because they supply the resources for the project (more on this in section 6.2).

In the various LEGO and Grundfos projects described in this thesis, managers have played important parts, often in steering committees, but sometimes also as sparing partners before important milestones.

Platform thinkers

Platform thinkers or employees experienced in platform development naturally have good starting point for identifying new platform projects. They might recognize the potential easier or they might simply be keener on applying platforms in different situations.

The fact that they might be keener on platform application can sometimes prove a downside in having platform thinkers designing new platforms, because they become too fond of the system in the platform, and thus become oblivious to what is valuable for the customer.

At LEGO, where the DesignLab unit is continuously developing new platforms (or platform components), it has been realized early on that without the constant participation of product designers and marketing persons (represented in management and in the design teams) many rational and logical brick assortments could be created only to remain unused.

The limited success of the Minifigure Accessories project and the Shapes project attest to this statement. Though the designed systems in these projects may be rationally and logically sound, they have gone too far from what was originally suggested by the designers and do not relate directly enough to what is valuable for the customers.

Similarly the Universal Elements projects were from the beginning designed so that it would be the product designers and not the platform designers who supplied the raw list of universal brick candidates. In this way it was prevented that bricks, which fitted nicely into the system where chosen before bricks, which were actually useful in the products.

I have mentioned lack of insight in platform-based product development several times in the above as an impediment for employees in making suggestions for platform projects, but it seems having too much insight also hampers good suggestions.

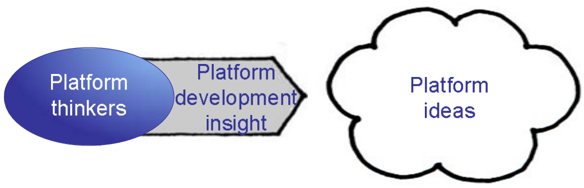


Figure 90: Platform thinkers can easily suggest platform projects, but such projects may be too systematic and thus lack customer value.

Platform thinkers should of course participate in platform projects but so should persons from other departments and persons with different backgrounds.

6.1.5. Conclusion

Based on the cases and some logical reasoning, I conclude that platform ideas may come from many various sources within a company.

The hypothesis, which I gave in section 3.2:

Platform project ideas are found and suggested by the product developers themselves, when they are designing new products.

is clearly not in accordance with findings from the cases.

The low-hanging fruits, which are often the first ideas, typically originate from insightful experienced workers based in product development or production departments. Good platform project ideas may however also be suggested by marketing and sales persons, but all of these persons require guidance or insight in platform-based product development to create or suggest platforms, which go beyond their own department.

Managers and platform thinkers may recognize the need for a platform or a platform opportunity, but they lack sometimes lack the required technical insight.

Finding platform project ideas must therefore be about prompting all of these people for ideas and then supplying what they lack (e.g. supplying customer-value to ideas from production departments or technical foundation to ideas from managers).

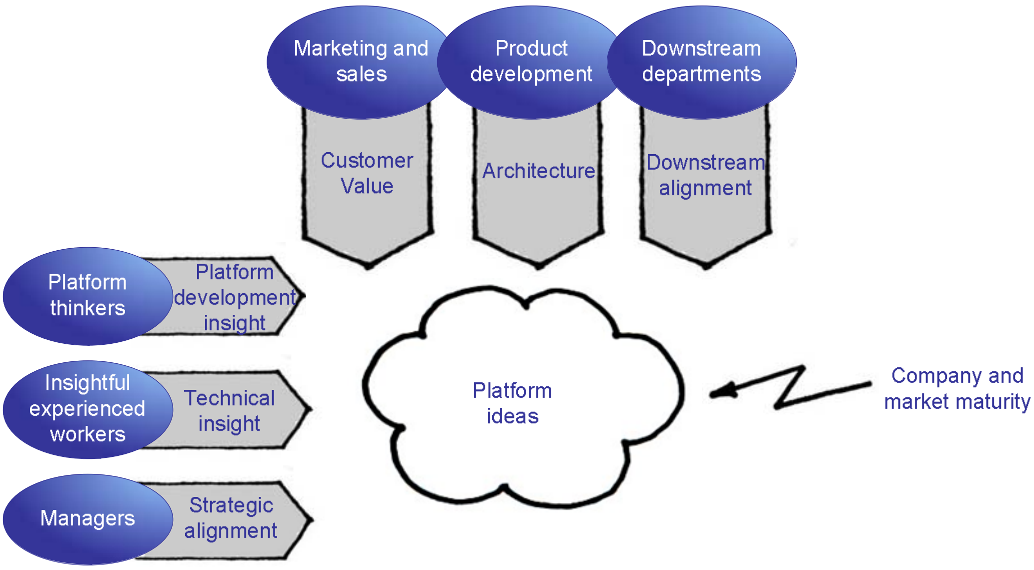


Figure 91: Platform idea generation. All of the listed persons and departments are capable of suggesting platform project ideas, although the best result is attained by combining inputs from several sources. The company and market must have a certain maturity to get good sustainable platforms.

Finally, if the company is not mature for platform development (i.e. there must be some degree of burning platform) or the product s-curve is not ready for it, no good sustainable platform projects will be proposed.

This conclusion is summarized in Figure 91.

6.2. Platform project proposal

To engineers or technicians modular structures can seem smart on their own. The fact that one component can be shared in different architectures, or that another component can be swapped out for a third in the same architecture can seem alluring and beneficent for many product developers or downstream employees.

To other employees or managers, this may however seem insignificant. A sales person will not think highly of modularity, unless that modularity can be related to sales parameters such as lower costs, shorter lead-times, or more customer options. Similarly a production manager will not think highly of modularity, unless it can be related to parameters such as fewer stocks, shorter assembly lines, and less tool investments.

To leverage the full potential of product platforms it may however be essential that these people recognize the importance of the platform and act accordingly by ensuring platform compliance.

To get broad acceptance of and commitment to a platform, it needs to be presented continuously for the various platform stakeholders – highlighting the benefits for these stakeholders and for the company in general.

6.2.1. Stakeholders

A very important part of initiating a platform project is to recognize the platform stakeholders and the roles they take related to the platform. The list of stakeholders is very similar to a list of stakeholders for a product development project (i.e. all of the persons who interact with the product throughout the product life and others who are affected by the product or must somehow ensure or prevent certain effects), but the roles these stakeholders take have changed.

Platform users

I use the term *platform users* for all of those employees within a company, who must behave in a certain way for a platform to work as planned. The benefits of platforms rely on employees to recognize platform potential and follow specified procedures (Figure 92).

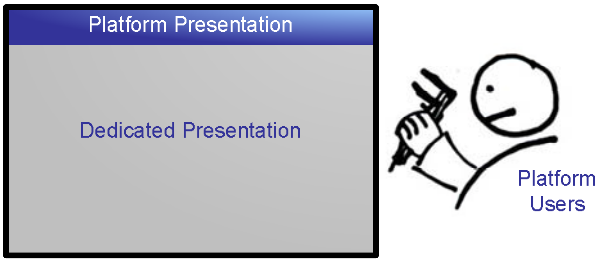


Figure 92: Platform users need to be informed correctly of the platform, if the success of the platform requires them to change behavior.

Marketing and sales persons must know when they are advertising or selling products, which comply with the platform, and what the consequences of selling those that do or don't are for the platform. They must also know how to behave when they receive an order for a platform product.

Product developers must be able to recognize a potential order for a platform product and know how they can manipulate the platform without disturbing the positive effects of the platform – as well as what the consequences of that manipulation are for the platform.

Downstream employees must be able to recognize platform products and components and know what processes to choose and which procedures to follow accordingly.

If any of these employees behave differently the planned benefits of a platform will not be leveraged and it is therefore essential that they do.

To encourage a new behavior it is important that all of the users realize the benefits of platform products for themselves as well as the company. Additionally, since platform products are based on well-known components, risks in general should minimize and the procedures and tools the employees use should be streamlined.

Marketing and sales persons will benefit from the platform indirectly. The downstream benefits and the benefits in product development will lower the total costs and lead-time of the products, and part of these savings should ideally be used to lower prices and delivery time of the products – making the products the employees advertise or sell more attractive. If the company neglects to reduce prices and delivery times and does not streamline the procedures, there is no incentive for these employees to behave in compliance with the platform.

Product developers will benefit from the platform directly. The benefits in product development as described in section 2.6 will reduce the resource load in product development, which again can be realized as direct savings, increase in output product quality or quantity, or reduction of lead-time.

Downstream employees will also benefit directly from the platform. The downstream rationalization opportunities described in section 2.6 can likewise be realized as direct savings, increase in quality or capacity or reduction of lead-time.

At LEGO presentations to the different stakeholders has been given continuously throughout projects since the Wheels project in 2005. Generally the product developers understand the benefits of new platforms and have realized the effects (documented in [Harders et al. 2008]), but the marketing persons and the downstream employees are still short of achieving this.

The marketing persons at LEGO lack the incentive to promote platform components based on a combination of two things. The effects of the individual projects are so limited that they are not significantly felt on the product level and the whole cost structure have changed so radically during the last four years that small or medium positive effects from platform projects fade when compared to the general development.

The downstream employees at LEGO lack a stable architecture because of the sourcing/outsourcing plans and it is therefore difficult to rationalize much. Furthermore the effects in production are heavily dependant on quantities and therefore these effects will not materialize until both marketing persons and product developers increase sharing.

At Grundfos presentations to the different platform users were given in the Mold Equipment project in a special workshop event. Consequently those employees who attended know how to behave in platform-based product development and can pass this knowledge on to their colleagues. Since Grundfos has already achieved some of the results from this project, it appears that this strategy has worked, although no detailed studies have been undertaken to verify this.

Platform decision makers

Besides the platform users the platform decision makers are the most important group. Platform decision makers are those managers who are not in direct contact with the platform or its components, but make decisions, which affect it. This might be the managers of product development, the managers of the various downstream activities, the managers of marketing and sales persons, and possibly the overall managers.

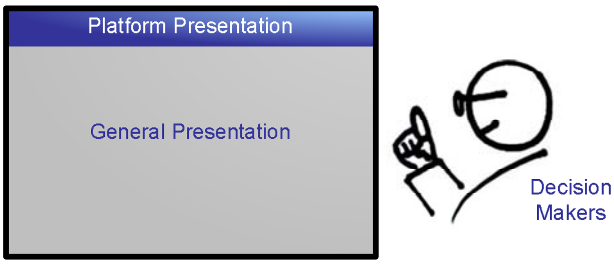


Figure 93: Platform decision makers need to be informed correctly of the platform for their decisions and dispositions to be in compliance with the platform.

These decision makers must ensure continuous resource commitment to projects, which have been approved. Since most platform projects require initial investments and will not pay back immediately, decision makers must keep supplying the resources as needed. No more resources will likely end the project, and all of the accumulated investments may be lost (Figure 93).

They must also ensure continuous commitment among their employees either by enforcing or rewarding the correct behavior.

To be able to do this, managers must get insight in the business case of the platform project. They must know what resources and investments are needed and when they are needed, and they must know the effects, which can be expected from the platform project and when they can be expected.

At LEGO correct behavior among product developers is already being enforced by the component frames and component costs described in section 4.1.2. These restrictions effectively ensure that product developers only develop new exciting bricks for their models when warranted.

For the individual platform projects at LEGO described in this thesis, managers have been brought into the projects as much as possible. The process of presenting the projects repeatedly for the managers not only aimed to prioritize the projects, but also to get management commitment.

At Grundfos, management is also informed of the platform project status and business case repeatedly throughout the projects, and here management is monitoring reuse of platform components, so that correct behavior can be enforced.

6.2.2. General presentation

An important part of making sure that a platform is accepted and implemented in an organization is the presentation of the platform project.

Platform users need special presentations, which describe how they can or must use the platform in order to leverage the benefits. These presentations will depend highly on the type of user (e.g. production employee or product development employee) and the type of platform. I will therefore supply no general templates or ideas for such presentations (i.e. these presentations are dedicated as stated in Figure 92).

Non-users like decision makers or employees, which will not use the platform directly, however, needs a general presentation of the platform.

To find out which components such a presentation should consist of, I have studied why some projects fail or are discontinued.

Typical causes of discontinuation or failure

In a company some projects will receive low priority compared to other projects and therefore be discontinued (formally or in reality when no resources are allocated). Other projects receive higher priority, but fail in the longer run.

The typical causes for eliminating projects or for failures are presented in this subsection based on the experiences from the various platform projects.

1. Costs outweigh benefits

   Sometimes it is the benefits that are marginal, and sometimes it is simply the economic scenario that does not add up. The LEGO Wheels project had this problem initially and new implementation scenarios had to be designed several times before the benefits outweighed the investments. A large number of other undocumented platform project suggestions and ideas have also been cancelled or discontinued because of too high costs compared to the benefits.

   A poor economic scenario is the most frequent cause for discontinuation of platform projects at LEGO. The platforms simply must pay off in some way, otherwise they serve no purpose. Because of this it is important to formulate economic scenarios early on in the project even before the concept is fully developed – because there is no reason to spend resources on a project, which will never pay off.

2. Poor or lacking platform concept

   Sometimes platform projects cannot be clearly explained. This has often been the result of poorly formulated problem statements, conflicting needs among the users, or misconceptions of the company strategy or capabilities. Unfortunately this has been the case with several platform projects at LEGO.

   An early Panels project discovered that even though several product development teams wanted to replace the panel components for their models, they eventually disagreed completely on why the existing panels were inadequate.

   Another project wanted to replace a production process, but discovered that this would change the appearance of a type of components, and this violated the company strategy.

   A poor platform concept is often recognizable or apparent in early presentations. This however requires a clear, understandable concept description. Projects can often continue through several milestones based on need and problem analyses, only to be abruptly discontinued when the concept is finally revealed.

3. Bad implementation plan

   Sometimes a platform project describes a goal, which is appealing, but fails to reach this goal fast enough. The LEGO Wheels project has almost collapsed several times due to arguments arising over component substitution. Could these components be substituted in reality and when should it be done?

   Another example is the Universal Elements project in 2007, which supplied components for model building so late, that they were assumed to be completely approved. When several of these components later where cancelled, the product developers became surprised and disappointed.

4. Lack of commitment

   Sometimes a platform project fails to gain commitment. This happens either, because there are too many projects, and this overwhelms management and reduces the efforts and focus on the individual projects, or because too many changes for the future users decreases the chance of full commitment and attention.

   Projects of course also fail to gain commitment, if they fail to comply with the needs of the product development projects. If the future users have no direct need for the platform or sees the platform as a hindrance instead of a tool, the platform has already failed.

   In the Platform Screening projects at LEGO we have clearly seen the effects of presenting and developing too many projects at once, and it is remarkable how many projects gain commitment in the years where few projects are suggested.

The general platform presentation should address these issues, and therefore there are four main components in my general platform presentation. Some of the presentation components may not be very comprehensive in the early parts of the platform development process.

Based on the many successful and unsuccessful experiences of presenting platform projects at LEGO, I will in the following subsections supply advice on how to present platform projects using these four components.

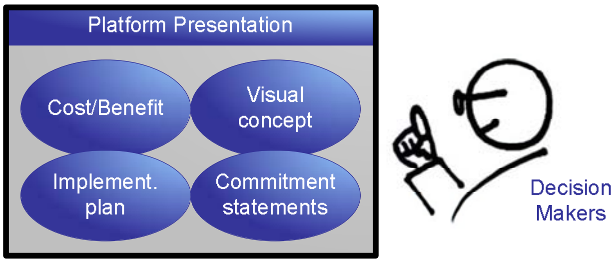


Figure 94: General platform presentation is made up of four components: Cost/benefits; Visual platform concept; Implementation plan; and Stakeholder commitment statements.

Costs and benefits

The purpose of product platforms is to increase sharing and thereby increase the efficiency of product development and manufacturing. Because the purpose is related to efficiency, it is important to look at the costs and benefits of a platform project.

It is very important to recognize that there are at least two very different costs to look at here. The first one is the costs of the ongoing platform project in resources and investments; these are the actual costs of platform project and should be evaluated against the benefits.

In the LEGO platform projects it has always been important to estimate the investments of the individual projects, because the economic incentives of the projects were never very big (excepting perhaps the LVM project). In the Grundfos projects however (and this complies with other cases from my colleagues) project investments were never seriously discussed; because there was a general feeling the project was inevitable. The experience from these projects however shows that at some point it always becomes interesting what the costs of the projects were.

The second kind of cost is the total costs of the platform-based products (i.e. variable and fixed), when the platform is implemented. This should in most cases be lower than for existing products (i.e. savings, which should be listed under benefits), although it may in some cases even increase.

Assessing the future costs of the products may prove difficult. The variable costs can sometimes be estimated based on the platform concept and the fixed costs may also be estimated based on development experience, but the true unknown factor is the sales volume.

Often the sales volume will be affected by a platform project, because the price, delivery time, and/or quality of the products change, how much it is affected remains however unknown, and in all of the platform projects I have participated in or heard about from my colleagues, this effect has never been estimated. In most projects volumes are assumed stable, although the very purpose of most platform projects is to raise sales volumes.

Cost estimates should still be included in platform presentations, but it should always be remarked that they are based on existing sales volumes and therefore mainly there for comparison (i.e. the variable costs part).

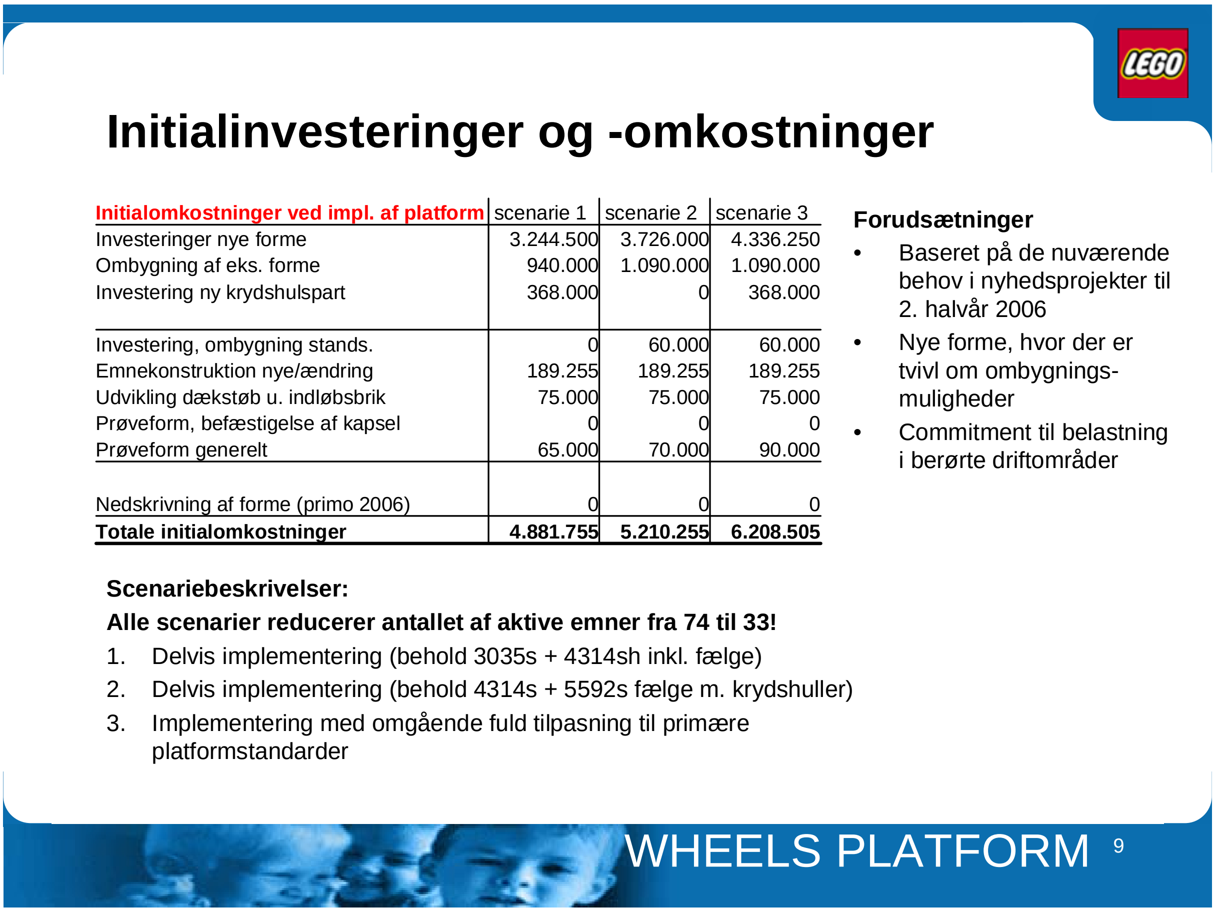
In the LEGO Wheels platform project we estimated the investments in molds and cost of components and calculated an economic scenario based on existing sales volumes of all of the LEGO Wheels components. Although simple in theory this actually required a number of different datasets to be combined and cross-referenced in order to include reordering of molds when they expired and also a gradual replacement of existing components with new platform components.

To be economically feasible the Wheels project had to spread out the replacement (and thereby the investments) of components over two years, and an expected platform life of 5-10 years made how implementation was handled economically crucial. Later LEGO platform projects have only confirmed that it is not enough to present figures of existing and future variable costs, because the investments for replacing the components may prove significant.

Another big disturbance in estimating the benefits of platform projects is technical uncertainties. Before a project is well underway, it can be difficult to truthfully answer all of the technical uncertainties. I have already explained why most technical uncertainties should be avoided, but some will always remain, and therefore there is likely to be different scenarios for the costs.

Different cost scenarios are typically presented as best case and worst case figures, but this simple method may not always be effective. Based on statistics (i.e. the chances of everything going good or everything going bad is slim) more professional estimates can be made, but even this approach may not be beneficial in some projects.

In the Wheels project, the best case and the worst case figures were very different. Management challenged us with the fact, that they were actually discouraged in committing to the project, when we were presenting worst cases. Usually product designers were very optimistic about their ideas, and then the managers could be skeptical. By presenting worst scenarios we were expressing doubts about our own project.



Initialinvesteringer og -omkostninger

Initialomkostninger ved impl. af platform	scenarie 1	scenarie 2	scenarie 3
Investeringer nye forme	3.244.500	3.726.000	4.336.250
Ombygning af eks. forme	940.000	1.090.000	1.090.000
Investering ny krydshulspart	368.000	0	368.000
Investering, ombygning stands.	0	60.000	60.000
Emnekonstruktion nye/ændring	189.255	189.255	189.255
Udvikling dækstøb u. indløbsbrik	75.000	75.000	75.000
Prøveform, befæstigelse af kapsel	0	0	0
Prøveform generelt	65.000	70.000	90.000
Nedskrivning af forme (primo 2006)	0	0	0
Totale initialomkostninger	**4.881.755**	**5.210.255**	**6.208.505**

Forudsætninger
- Baseret på de nuværende behov i nyhedsprojekter til 2. halvår 2006
- Nye forme, hvor der er tvivl om ombygnings-muligheder
- Commitment til belastning i berørte driftområder

Scenariebeskrivelser:
Alle scenarier reducerer antallet af aktive emner fra 74 til 33!
1. Delvis implementering (behold 3035s + 4314sh inkl. fælge)
2. Delvis implementering (behold 4314s + 5592s fælge m. krydshuller)
3. Implementering med omgående fuld tilpasning til primære platformstandarder

Figure 95: A slide presenting the initial investments in the Wheels platform project. The three scenarios listed in the table are based on different component replacements and different timing. General assumptions are listed on the right. This layout was adopted after the best-case/worst-case presentation had been rejected.

In the Grundfos Molding Equipment project this was even more profound. When going through the worst scenarios the outcome of the platform project was sincerely questioned, and although all of the product developers believed that the platform was good and beneficial, nobody would commit to anything more than the worst case, where the platform would actually be an economical disaster.

Wiser from the experience, we adopted another strategy in the Grundfos Test Bed project. Here we presented the most likely economical results and results that were based on certain explained contingencies.

Based on these observations, estimating best and worst cases in platform development can be hard. Because of the technical uncertainties and the sales volume uncertainties, the best and worst cases can be far apart ranging from disaster to incredible success. Best and worst case estimation may often be beneficial in cases where these figures are more closely comparable, but can seem pointless when there is too much difference. Product developers should instead offer the most likely results based on reasonable assumptions (as in Figure 95).

Another measure for efficiency is time.

Again there is the time invested in doing platform activities. Ideally this should be done when it does not influence the manufacturing lead-time, either by undertaking these activities in low-intensity periods or by having other employees undertaking them, if this is possible there is no lead-time invested in the platform project but rather an added cost. If it is not possible however there is a genuine investment in lead-time in the platform project, which should be evaluated against the benefits of the project.

The other time assessment is the time it will take to manufacture (i.e. develop, produce, and assemble) the platform-based products in the future. In some cases development of these product variants will disappear completely, because all components are developed beforehand, and thus the manufacturing time will have no fixed, but only variable elements (as in the Wheels project).

Other benefits cannot be quantified or are not directly comparable to the costs. These include general reduction in risks and increase in cost and lead-time estimation of the new products, quality improvements, increase in capacity, alignment with specific strategies, increased knowledge sharing and learning opportunities and other benefits described in section 2.6. Still listing these in a cost benefit analysis and assessing their magnitude seems necessary, since they sometimes are the primary benefits.

Visual platform concept

Because of all of the unquantifiable benefits communicating the costs and benefits of platforms is often not simple. Many assumptions are made and figures are estimated or guesstimated in order to create something, which can be compared.

The purpose of the lists of costs and benefits was to provide non-technicians with a view, which was easier to overview and understand, but due to the many assumptions and guesses, this view is often either incorrect or highly complex.

In the LEGO platform projects this has been very evident. The lists of hard and soft benefits for a platform project did often not suffice. Many questions would arise, when the benefits were presented, on why specific assumptions had been made and why the results were as they were.

Therefore we started also presenting descriptions of the platform concepts. This was however not easy, because platforms are abstractions and cannot be completely visualized.

We could show the bricks (or examples of bricks), which would be created based on the platform, and we could sometimes show some of the rules that applied to these bricks.

Elements which have been utilized in these concept visualizations include:

Architectures or sub-architectures were used in the discontinued Creature Building project 2006 and the discontinued Creature Building project 2007, see Figure 96. Architectures are typically visualized with block diagrams, where each block represents a generic placeholder module, which can be swapped and replaced by a number of different standardized modules. For increased visualization pictures of sample standardized modules can be inserted in the architecture diagram for easier recognition as shown in Figure 96 (left).

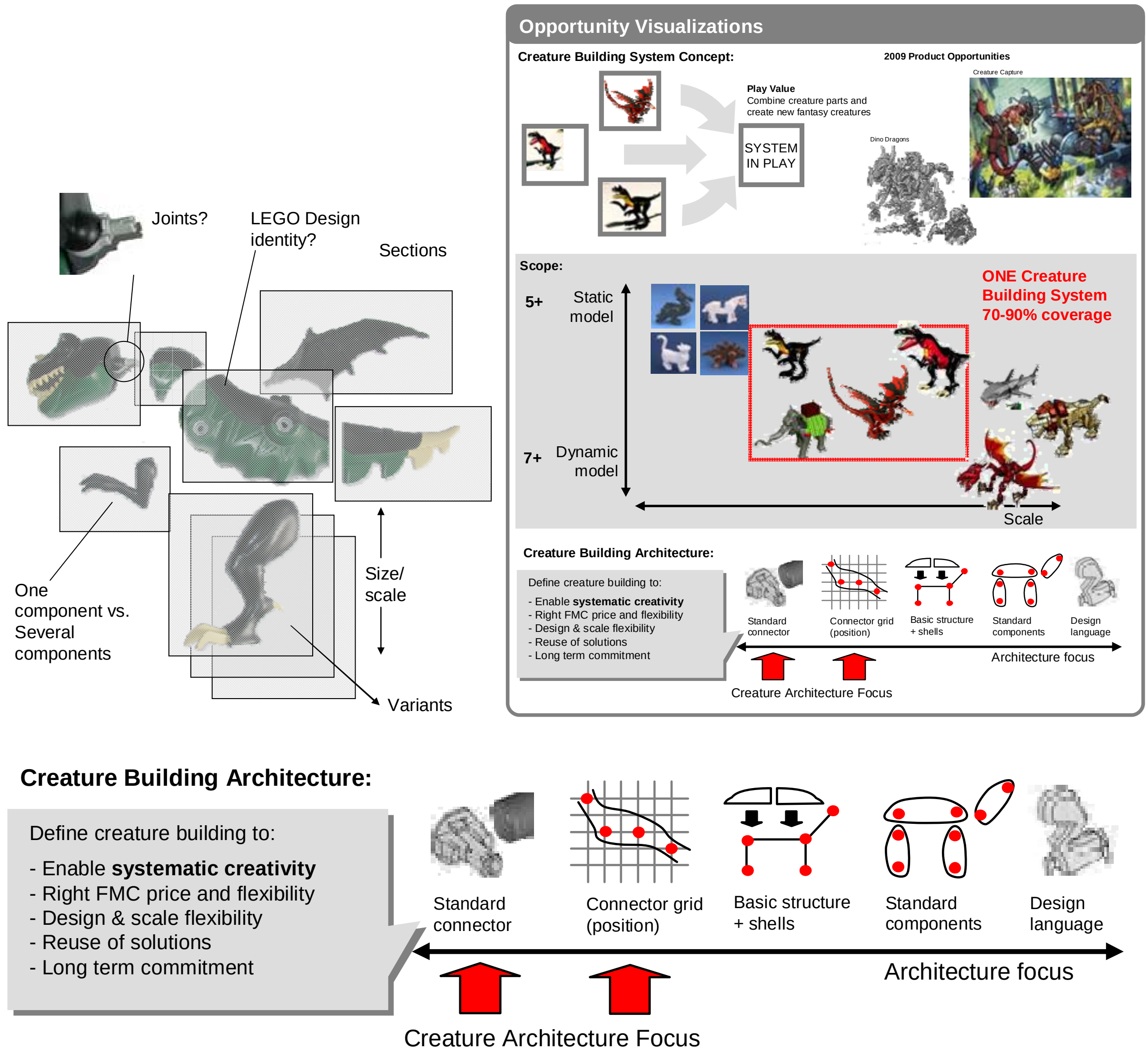


Figure 96: Element from the discontinued Creature Building 2006 project (left) and full visualization of the discontinued Creature Building 2007 project (right) with close-up (bottom). A traditional architecture visualization was the primary element of the 2006 presentation. The 2007 presentation was not as strict and showed the intended common architecture level on a generic scale of architecture levels (bottom).

A more advanced diagram was used in the presentation of the Creature Building project 2007 shown in Figure 96 (bottom). Here an axis of different architectural abstractions is presented ranging from sharing of only connectors (i.e. interfaces) through sharing of placement and positioning of components to full sharing of standardized components and architecture.

Another visual element is to show an example of a generic placeholder module and a set of standardized modules, which fit this placeholder module as shown in Figure 97. This kind of element is beneficial when there is no common total architecture (as in the Walls & Windows project case) or when it is very complex.

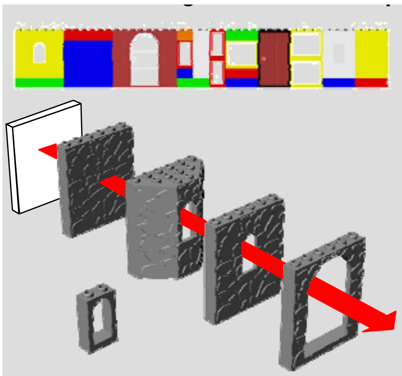


Figure 97: Element from the Walls and Windows 2006 project. A generic placeholder module and four standardized modules are shown. Each of these four modules complies with the agreed-upon common interfaces. Above the modules a line-up of components is shown to illustrate the size of the components and how they can be combined.

In many of the LEGO projects pictures of the components are shown – either as a sample selection of components because the assortment is not fixed, or as the total assortment if it is fixed. This was done in the Wheels project and in the discontinued Functional Blocks 2006 project (Figure 98), the Frames 2007 project (Figure 98), the Panels 2007 project (Figure 99), and the Shapes 2007 project (Figure 99).

Some of the LEGO projects were not as well founded on a definite component design, and in these projects visual elements centered on model opportunities (i.e. products, which the platform would make possible or easier), this was especially true for some of the projects, which were later discontinued – such as the discontinued Frame Elements 2006 projects and the Mini Joint 2007 project.

As can be seen from the platform presentations the general approach to visualizing the platform concept has been to illustrate a need either by illustrating a gap in the existing assortment followed by pictures of the components, which would fill this gap, or by the new components and the product opportunities they would result in, or both.

The Grundfos projects were visualized by the same means. The Molding Equipment project was primarily visualized with pictures of the platform components (section 4.2.1, Figure 64) and the products which could be build using these components (section 4.2.1, Figure 62). Whereas the Test Bed project was visualized with pictures of the platform architecture (section 4.2.2, Figure 68) and a sample product which could be build using the platform components and architecture (section 4.2.2, Figure 67).

The visual element in a platform presentation is very essential – especially when describing the platform for employees with other backgrounds or positions. It is often this element, which should receive the most attention, and therefore needs much preparation and thought.

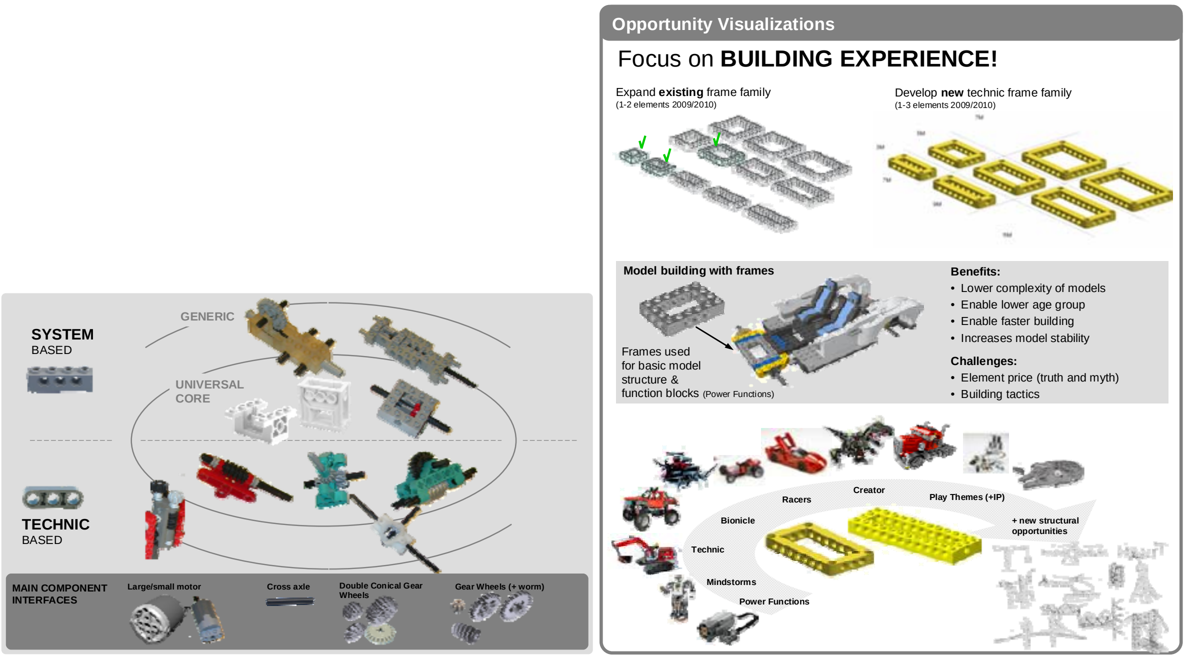


Figure 98: Element from the discontinued Functional Blocks 2006 project (left) and full visualization of the Frames 2007 project (right). Both presentations showed complete suggested assortment of bricks, although the Frames project presentation showed the bricks in a size-grid.

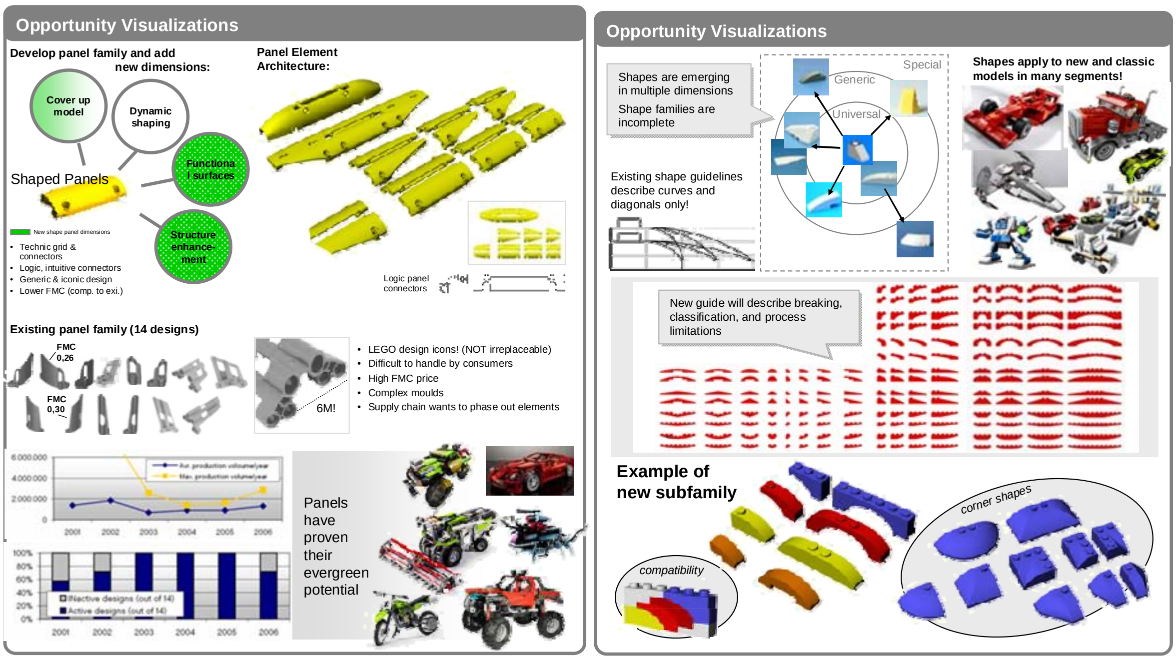


Figure 99: Full visualization of the Panels 2007 project (left) and full visualization of the Shapes 2007 project (right). The Shapes project only showed a portion of the suggested bricks, because the total shapes assortment is virtually limitless.

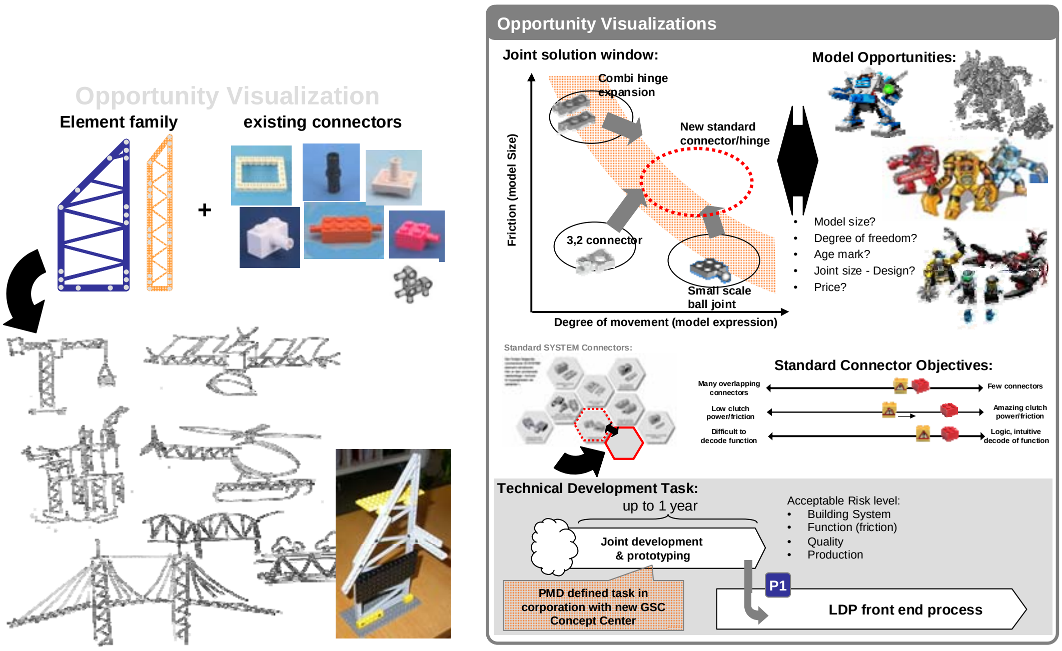


Figure 100: Elements from the discontinued Frame Elements 2006 project (left; unrelated to the Frames 2007 project) and full visualization of the discontinued Mini Joint 2007 project (right). Both presentations were focused on the products that the platform would make possible.

Implementation plan

An often stated problem with platform projects is that they are about future activities and projects, and therefore have problems competing with present projects over resources, if resources are scarce. To get around this problem platform projects need to be relevant to other ongoing activities.

A frequently occurring element in platform presentations is implementation plans in the form of schedules (e.g. Figure 101 and Figure 102). This corresponds nicely to the needs described in section 6.2.1; it is essential that managers and other employees know when the various costs of the project will be needed and when the various benefits can be leveraged.

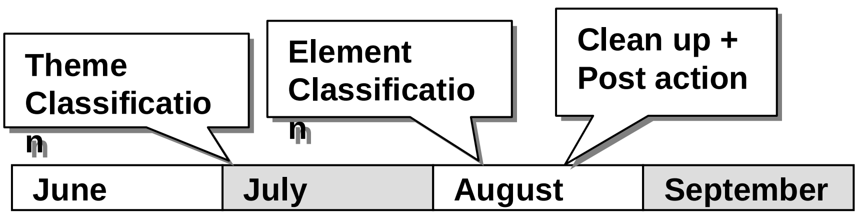


Figure 101: Implementation plan for the Minifigure Classification 2006 project (Closeup of Figure 47).

The LEGO platform projects are generally implemented after a schedule, which ensures that they will be ready and applicable for the annual product novelty projects. This does not mean that LEGO platform projects end when the novelty projects begin, but rather that the first platform components must be ready for evaluations and tests, when the projects need them. The platform components will not be finally manufactured long before the other components, which go into the annual products. Several successful platform projects (e.g. the Wheels project, the Walls & Windows project, and the Panels project) have had implementation plans for a series of years, where each year new platform components are being made ready for evaluation and tests.

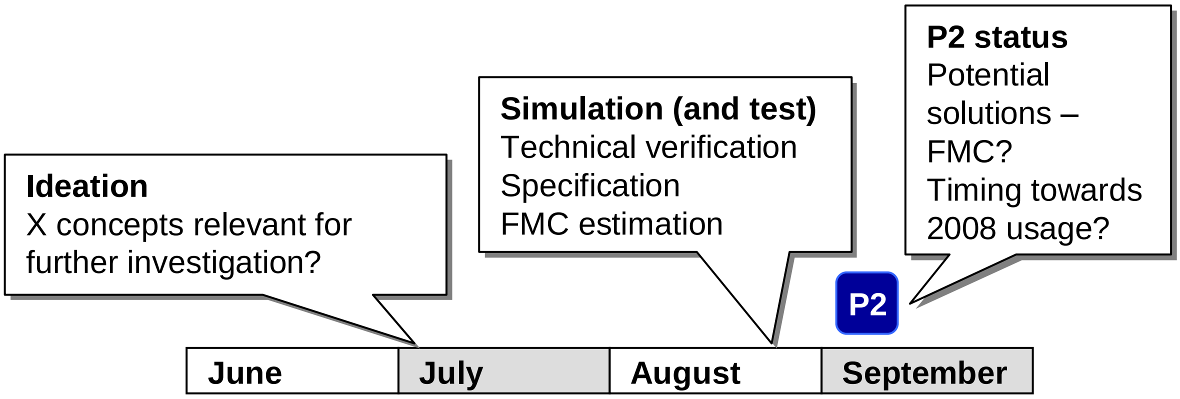


Figure 102: Implementation plan for the Ball Joint 2006 project (Closeup of Figure 49).

In companies, which have a less strict annual program for launching new products, like Grundfos TC, the implementation plans for platform projects are not as easily determined. Still, to get the necessary commitment and attention from the company, experience shows that it is beneficial to tie platform implementation plans with product launch plans.

In the Grundfos Molding Equipment platform project only the collapsible core platform component was needed immediately in product development projects and therefore it got much more attention, and was eventually the only platform component in that project, which got implemented.

In the Grundfos Test Bed platform project the platform project was intentionally tied to a number of new projects aimed at supplying Grundfos Service Centers with new inexpensive test beds and to the start-up of a new test bed manufacturing plant. This platform therefore received a lot of attention.

The ties between platform development and present product development will be further examined in section 8.

Stakeholder commitment statement

A platform project, which has general commitment in the organization, is much surer to succeed than one, which has not. A platform project, which has still has no commitment, might only be an uncontested idea of one or a few employees, which may be proved poor by more experienced or knowledgeable employees, while a platform project, which has commitment has the support of these key employees.

Many of the LEGO presentations feature small figures, as in Figure 103, which show the stakeholders who support the platform project. This not only shows the magnitude of the project, in terms of which groups are affected, but also confirms that these groups support the platform project.

In the Grundfos projects a less visual approach has been used. Instead of showing support by listing design teams, the most (or one of the most) experienced senior product developer was given the role as presenter. This indicates that this key employee is committed to the idea and therefore has an equivalent effect.

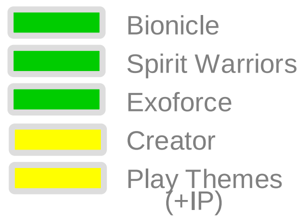


Figure 103: Example of a stakeholder commitment statement taken from a platform presentation at LEGO. This project influences the five mentioned product development teams. Green means that the team is fully committed and will participate in the design of the platform or parts thereof. Yellow means that the team accepts the platform and will use it, when it is ready, but will not participate in the project.

6.3. Elimination race

Not all platform ideas evolve into good platforms. Some platform projects discover that the planned benefits result in some unavoidable bad secondary effects and other projects simply lack the timing or the organizational commitment.

Before assigning resources for platform projects, all potential platform projects should be carefully evaluated and prioritized as stated in the introduction of this section and by [Otto et al. 2007]. But while we need to evaluate the projects before we start the projects to prevent bad investments, we also need to start the projects before we can evaluate them. The result is a process, where the projects are gradually developed and evaluated in a process similar to the concepts of set-based product development [Morgan et al. 2006].

6.3.1. Parallel development of platforms

The basic assumption, which this concept of developing platforms in parallel builds upon, is that a company in general has too few resources to pursue every proposed platform project candidate. This means that the projects must be prioritized, so that it is not simply the projects, which are proposed first, which get the resources.

To be able to prioritize projects, they must be made comparable, but platforms, which are described with very different amounts of detail, are not very comparable. Therefore the platform projects needs to be developed in parallel, so that they will have roughly the same amount of details.

This is exactly how it has been done at LEGO. Each year a number of platform project candidates are proposed and prioritized simultaneously as described in section 4.1.3. The projects are gradually developed and evaluated and only the best projects are continued. The process follows the steps indicated in Figure 104.

In the LEGO projects we learned that developing platform projects in parallel held several additional benefits as well as some drawbacks.

When developing the platform projects in parallel we were able to reuse presentation templates. This naturally saved us some time, but it also meant that we needed to agree on what parameters and figures were needed in platform presentation in general in order to create the template. The results of these discussions were presented in the previous subsection.

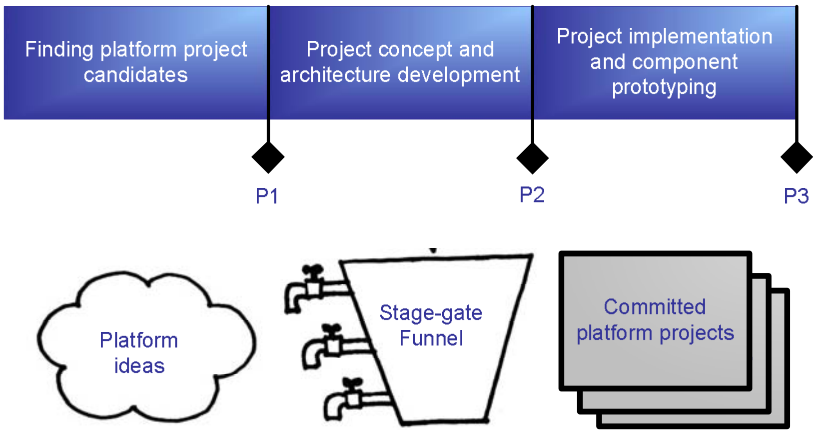


Figure 104: The parallel development process concept at LEGO. Platform candidates are gradually developed and evaluated. The best projects continue and are given commitment. Committed platform projects are only discontinued based on technical or performance measures – not based on prioritization.

Much more importantly, when presenting several platform projects in the same presentation session using the same template, the concepts were communicated much easier. Decision makers and other stakeholders would recognize the template and faster overview and find the information they needed in subsequent presentations.

When comparing the different presentations, the audience was able to ask additional questions (e.g. why does this presentation not show an investment figure, when the others do?), because they could see what was not presented in the individual presentations. These observations would likely have gone unnoticed if there had been no comparison, since the audience would have assumed omitted things to be forthcoming in a later presentation or simply not have though of it. Now they were also able to comment the development progress, when comparing the projects.

Also bundling presentations in a few sessions, where all platform projects were presented, created more attention (physical attention at the meeting as well as general attention in the organization) than having multiple small sessions would have.

The drawbacks were primarily related to the fact that the platform projects were different in nature and therefore equal treatment was not always good.

The initial platform ideas ranged from simple need identifications to ready-made concepts, and bringing these ideas to the same level of detail was difficult and highly favorable for the less detailed projects. Several times all of the available resources were spent on improving relatively undefined projects, which only held small chances of succeeding, when these resources could have been spent more effectively on the better projects. As a result early prioritization was intensified and carried out internally in DesignLab even before platform projects would be presented to decision makers and stakeholders.

Also some good projects simply needed more time than others and fixing these projects to a set timeframe actually prevented some projects in succeeding, because they were discontinued due to poor chances of realization within the timeframe. It is currently being discussed in DesignLab if some projects can be given a two-year pre-development timeframe, or if such projects should be handed over to other parts of the organization, which are focused on long-term development projects.

Grundfos TC is not as experienced in running formal platform activities in parallel. The Molding Equipment project and the Test Bed project were undertaken as pilot projects before a general platform rollout. When this general rollout hopefully begins late 2008 or early 2009, several platform projects will be running in parallel.

The purpose of running platform projects in parallel at Grundfos will initially not be to ease prioritization. Since each business unit will undertake one project, these projects are independent, and since all of the projects have separate resources (the LEGO projects must be undertaken by the same experienced LEGO employees, whereas the Grundfos TC business units each have their own experts with limited insight in other areas) and must return the investments, all projects can be undertaken simultaneously – in theory. The primary reason for developing platform projects in parallel at Grundfos is that projects cannot wait; the burning platform forces the company to act soon.

Two factors will likely prevent a complete parallel rollout. One is the limited expertise in platform-based product development. The technical experts cannot create the platforms on their own, but will need facilitators or guides to do so (as explained in section 6.1.4). These are limited.

The other factor is that too many simultaneous projects will be to complex to manage and overview. If too many projects run in parallel, some are bound to receive too little attention from decision makers and stakeholders. This last factor has been experienced in the LEGO projects.

6.3.2. Platform project and presentation evolution

As stated before, platforms must be presented multiple times for stakeholders and decision makers to gain commitment in the organization. In section 6.2 I presented the primary elements of such presentations, but as the projects evolve so does the presentations.

The overall platform project process described in Figure 104 shows three different stages of the platform development process at LEGO.

1. Stage one is about finding platform candidates or ideas and sorting out the worst, the remaining are then presented in the platform project proposal. At this stage the platform projects are most often simple ideas, which seem useful, or identified needs, which must be solved.
2. Stage two is about exploring the concepts and determining their scopes. During this stage the concepts evolve radically and the more vague concepts must meet reality. It is primarily in this stage that the projects are prioritized and continued or discontinued (see Figure 105).
3. Stage three is about determining the exact platform components, testing them, and implementing the platform (i.e. making sure that it has commitment and will be used in the future). Projects are not normally discontinued at this stage, but individual platform components may be cancelled because of poor tests or transition schedule.

After these three stages platform brick-components are manufactured as other LEGO bricks, although the platform project team may monitor and manage the process.

Important milestones in the LEGO process, where platform projects are prioritized, are:

1. During the initial stage, before P1, platform projects candidates are collected and sorted in DesignLab continuously. Many projects ideas never make it past this stage.
2. At the P1 milestone, platform projects are presented as rough ideas and needs. Qualitative costs and benefits are presented and so are a preliminary stakeholder list and sometimes a visual concept.

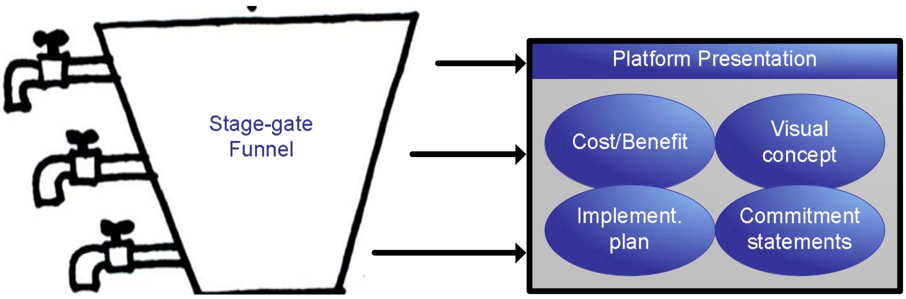


Figure 105: Stage two in the overall development process (Figure 104) is an elimination race, where the projects are gradually developed and evaluated. Good projects continue down the funnel, not-as-good projects are discontinued and taken out of the process.

3. Between P1 and P2 there has been put an extra milestone, because the projects evolve radically and the least good platform projects can be discontinued. An estimated quantitative costs and benefits analysis must now be ready and stakeholders must have given commitment.
4. At the P2 milestone, the final number of platform projects is approved and each project receives a budget and commitment. An implementation plan must be ready and based on this a fairly certain budget (costs and benefits) for the project. The architecture (documented in the visual platform concept) must also be finished.

6.3.3. Conclusion

Based on the cases (especially the LEGO cases), I conclude that platforms can indeed be prioritized by comparison with other platform project ideas.

This was exactly the hypothesis, which I gave in section 3.2:

Platform project ideas are prioritized by comparison with other platform project ideas.

This explanation is however very vague and immediately creates a new question. How can platform projects be compared?

The LEGO cases show that, when platform resources are limited and it therefore is necessary to prioritize projects, the projects can be compared and evaluated against each other in shared presentations. To be able to compare the projects, they must be on the same level of detail and a number of platform aspects must be presented.

These required aspects include:

1. Costs and benefits. Quantified when possible, but always listed and described qualitatively.
2. Visual platform concept. Preferably with platform components and elements, but otherwise as visualization of needs or problems.
3. Implementation plans. Describing how the changeover will be handled.
4. Commitment statements. With lists of those users and decision makers, who support the project.

To get an even better assessment of those projects, which are hard to prioritize, they can be pre-developed iteratively and presented repeatedly.

The overall platform project scheme will therefore consist of three stages. The first one is the gathering of platform ideas as described in section 6.1. The second one is the elimination race described in this section, where the platform projects get pre-developed and presented.

And the third one is the true platform development stage, which will not be described in detail in this thesis (Figure 106).

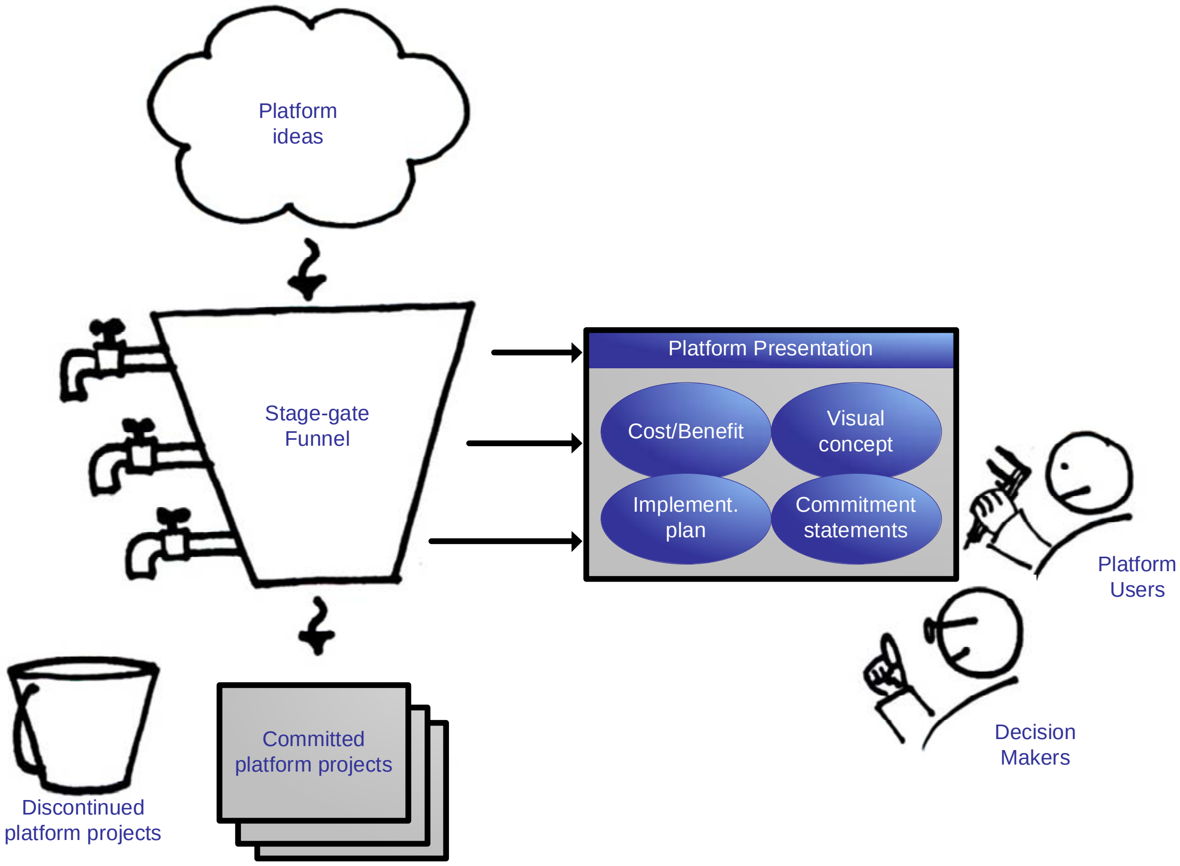


Figure 106: The overall platform development process consists of three stages. The first one is the gathering of platform ideas as described in section 6.1. The second one is the elimination race described in this section, where the platform projects get pre-developed and presented continuously. And the third one is the true platform development stage.

7. Maintaining Platforms

This section aims at answering the research questions stated in section 3.3 or confirming the hypotheses from the same section.

How do we monitor the performance of platforms?

and

How can additions and revisions be carried out without undermining the platform foundation?

State-of-the-art literature generally does not describe how platforms are maintained, and therefore answers cannot be found there.

However, platform maintenance is closely related to platform responsibility, because having responsibility for a platform assortment means making sure the platform performs as intended.

[Sanchez 2004] briefly touches upon the area of platform responsibility, when he mentions a product architect. The product architect is a product developer, who is especially dedicated to creating product platforms (or product architectures). This person is responsible for the shared components in the product assortment and for the process of deriving product variants from the generic platform concept.

This does not answer how to monitor platform performance, it does however indicate, who does so. Neither does it answer how additions and revisions are carried out, but it indicates that either must all product developers be product architects (thereby making the title meaningless) or only a select few product developers can do this.

[Kristjansson 2005] deals with assessment of platforms performance so his work partly answers the first research question. The assessment tool he has developed is however primarily targeted against large companies and large platforms, and is simply to extensive to apply unmodified in the LEGO and Grundfos cases. Furthermore, the assessment is based on interviews with platform stakeholders, where these persons assess different factors related to the platform maintenance. Our experience at LEGO however, shows that platform stakeholders and platform owners do not always know how a platform *really* performs [Munk 2009].

As a consequence, I present here another approach for assessing the performance of platforms. This approach is based on the case studies described in section 4. In these case studies I have gained some additional insight into the area of component management, which I will present in the following section. I realize that component management is likely only one part of platform maintenance, but the case studies offer little additional information besides on component management.

7.1. Component management

When components are not dedicated to specific products, they are no longer merely product parts, but are products on their own. Platform development designs and manufactures these products based on forecasts or orders from product development projects. This is in many ways similar to the way the company manufactures real products.

A key aspect of product management is monitoring of various product performance-data in PDM and ERP systems. Similarly we should monitor equivalent component performance-data in component management.

LEGO has manufactured generic product components for more than 50 years now, and should therefore be experienced in component management, and not surprisingly, LEGO is monitoring and managing component data in the same systems as they are monitoring products. Traditionally, LEGO has however not given as much attention to the individual components as to the products.

Following the economical problems described in section 4.1.2, the company has concluded that many of their problems relate to, how their component portfolio was managed (and monitored) in the past decades. The already-described initiatives (i.e. component frames, component classification, and component costs) are all examples of how LEGO now has learned to give more attention to their product components.

7.1.1. Platform documentation

Before answering the research questions, I will briefly describe the needs for platform documentation based on the LEGO platform projects.

Creating and maintaining platform documentation is an important part of platform maintenance. Without proper documentation platform users and decision makers have no reference when questions arise. This is the primary purpose of platform documentation: To serve as a reference for platform users (and sometimes decision makers) in how product development is carried out in compliance with the platform.

The second object of platform documentation is to provide data concerning the platform, which will enable a continuous evaluation of the platform performance.

The third object of platform documentation is the documentation of the platform project process. Since platform projects often are lengthy, a logbook of the project is beneficial when project members are added or replaced or for referencing in the same or similar projects. This project process documentation will not be further discussed in this thesis.

The three documentation objects are shown in Figure 107.

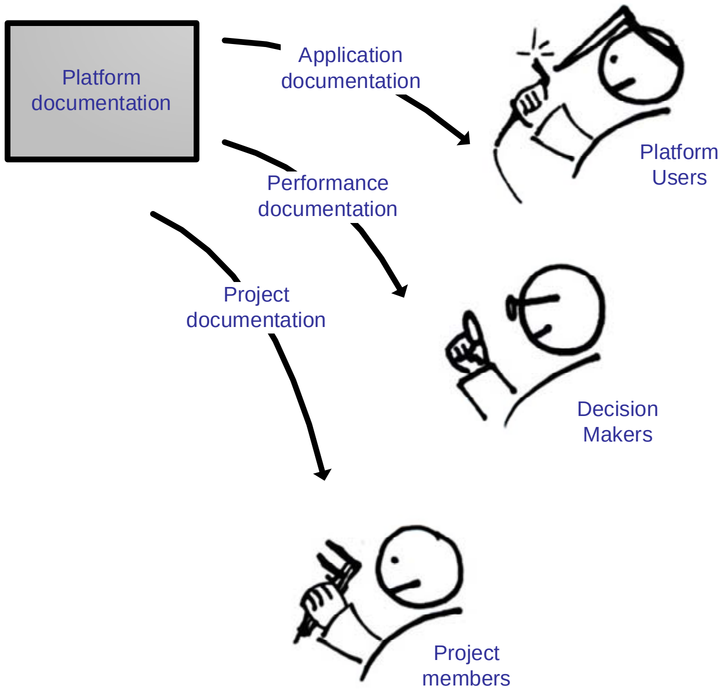


Figure 107: The different kinds of platform documentation. Application and performance documentation is detailed in the following sub-section.

Documentation data

As stated in the above, there are at least two kinds of data, which should be available for every platform and platform project: Data needed by platform users and data needed for platform evaluation.

For platform users there are a number of needs.

1. A general understanding of the platform.

   This can be provided by the general presentation presented in section 6.2.2. Naturally the latest presentation should be saved for future referencing.

2. A clear overview of the platform components.

   This can be provided by the PFMP described in section 2.3.3 or simple boards of available components like the LEGO brick library in mentioned in 4.1.3 or the more dedicated charts of bricks from the various LEGO platform projects (e.g. Wheels project and Shapes project).

3. One or more diagrams of the architecture.

   This can be created in various ways as described in section 2.3.3 with function- or structure-based diagrams. The purpose of these diagrams is primarily to show the relations of the different components (i.e. which components connect directly function- or structure-wise). As well as give an overview of the product build-up.

4. Placeholder module design guides.

   For each open generic placeholder module in the architecture, which makes it possible to design new components fitting into the architecture in the future, a design guide is needed. This guide should describe the necessary requirements and restrictions in terms of interfaces and general purpose for designing components, which fit the specific generic placeholder module.

This type of module design guide is used extensively at LEGO, where many brick sub-groups have individual guides on how to design components within the group.

5. Standard module application guide.

   For each pre-designed standard module available in the platform setup an application guide is needed. This guide should describe the interfaces and purpose of the specific standard module, which will enable designers to incorporate it in future products.

   Such module descriptions were created in the Grundfos Test Bed platform project.

   Another way of approaching module application guides is through classification. This is described independently in the following sub-section.

The following data should be recorded for platform evaluation:

1. Usage data.

   For all standard modules, it should be recorded which products they are used in and how many products they have been used in totally. This kind of data is incredible useful and is often the outset of platform modifications or the creation of a new platform. The purpose of making standard module is to be able to apply a common module in multiple products, and this data will therefore show how successful the different standard modules have been in product development.

   In more advanced platform cases, architectures and generic placeholder usage can also be recorded.

   In most of the case companies referred to in this thesis, usage data is not recorded specifically. It is often possible to identify specific modules in product BOMs, but no usage statistics is kept separately. At LEGO, usage data has been created manually for many platform projects, to document the general applicability of components, but this kind of data remains hard to retrieve.

2. Production data.

   For all standard modules the production quantities should be recorded. These numbers can be compiled from the usage data described above and the production quantities of products, which are usually recorded. This data effectively shows how successful the platform has been in creating a foundation for rationalization in the downstream activities. Ideally standard modules should be produced in much larger quantities than products, if they are used correctly.

   Because of the modular structure of LEGO products (and because each module consists of one produced part), this data readily exists, and is highly useful.

Component classification

At LEGO all components are classified. They are LEGO bricks, Technic bricks, or Duplo Bricks. They are also Universal, Generic, or Special. They each belong to a sub-group (e.g. Wheels, Windows, Minifigures), some are subject to special license agreements, and some are restricted to older age groups. All of these classifications effectively guide the designers, when they are selecting which components to use in new LEGO products.

Besides these very user-oriented classifications, LEGO has a number of info-codes, which guide maintenance, planning, and production of the components. Examples of things that are registered are:

1. The component is discontinued: Molds are to be recycled or destroyed.
2. The component will be changed, when a new mold is ordered.
3. The component is being substituted with another component.
4. The component is restricted to certain colors and materials.
5. The component has a certain flaw or limitation.
6. The component has been produced in another color, the mold exists.
7. The component has a new number (used with design changes)
8. The component is new. Time schedules need to be verified.

A lot of this information is useful for designers as well as employees in downstream functions, and license restrictions and age restrictions are also registered with info-codes.

All of these classifications are not directly related to platforms or platform maintenance, but are more appropriately related to traditional production and maintenance management. Similar classifications can however be used for platform maintenance.

When components cease to be simple components and are instead regarded as standardized modules, the ownership of the components shifts from the production department to the platform (or product development) department. The platform (or product development) department then has the responsibility of keeping the component updated and applicable.

In the Minifigure Accessories platform project at LEGO, we tried to allocate the responsibility of each component to the primary user. This meant for example that all of the spear components, sword components, and armor components would be associated with the LEGO Castle group and all of the police cap components, firefighter helmet components, and briefcase components would be associated with the LEGO City group. These groups would then have the responsibility of maintaining an adequate and updated assortment of components and would have to replace rather than add to their assortment. Other groups would still be able to apply these components, but would not influence their design and color variations.

The purpose of this proposition was to reduce the number of slightly outdated components, which could not be changed because they were used in multiple products.

Another classification proposition in the Minifigure Accessories project was the introduction of a time-fixed discontinuation for certain new components (symbolized with a time-bomb). The reasoning behind this proposition was that some components were never meant or designed for use in more than one or two products, and instead of keeping these components for the future (as is the default with LEGO components) they should be automatically discontinued, when the products, they were in, were discontinued.

Unfortunately these classification propositions died with the discontinuation of the Minifigure Accessories project. The project failure was due to the common misconception that it was a simple clean-up project (which it was not), and that the component ownership proposition was not supported by economical scenarios and therefore received little attention among platform users or decision makers.

The project propositions have however not died out entirely in DesignLab, and it is likely that a future Minifigure Accessories revision will complete some of these ideas.

I believe that the increased classification is necessary to facilitate continuous maintenance, updating, and clean-up of components as well as correct application of standardized components in product development, and these ideas are built upon in the following subsections.

7.1.2. Component revisions and additions

Platforms must continuously be updated and available for product development, or else they will quickly become irrelevant and forgotten. The continuous evolution of the company products however, requires the platform to evolve also. This was described in more detail in section 3.1.

As long as the platform project is running and the team, which designs and implements the platform, is still functioning, it is obvious that they have the responsibility for making sure that the platform is updated along with the product evolution.

In the long run however, the company must either keep the project running indefinitely or hand over the long-term responsibility of the project to other employees.

At LEGO, platform project teams are cross-organizational staffed by combinations of employees from DesignLab, the product development teams, and downstream departments depending on the specific platform project. When the platform project closes however, responsibility of the component platforms is handed over to the DesignLab unit alone.

DesignLab first and foremost ensures that components are available and manages the components on a daily basis. The actual updating is not carried out as part of the normal working routines, because DesignLab has no decision-making authority. DesignLab does however, suggest new platform projects and among these platform projects suggestions are updates of older platform components. Additional components or replacements are also suggested by the product development teams, but then it is DesignLab's responsibility to ensure compliance with the platform.

At Grundfos TC the platform teams are based on the independent product development units and possibly complimented by downstream representatives. When a platform project closes at Grundfos TC, the responsibility of keeping the platform updated and available remains within the same product development unit.

It is planned that the Grundfos product development units will add new platform components to the platform as new products are being designed, but this process has not started yet due to the relative newness of the platforms.

Adding new standard components

LEGO product designers are well experienced in adding platform components to existing platforms. It is a basic rule in component design at LEGO, that there are certain overlaying rules that define LEGO components, and that these rules have to be followed when adding new bricks to the system. Most designers know the basic rules by heart, but a number of guides and references exist to help the less experienced designers. New components are evaluated by peers after their compliances with these guides and references and modified accordingly.

LEGO components are by default treated as standardized platform components ready for reuse and sharing across the product assortment.

Compared with most of the other case companies this product designer behavior is remarkable. Often product designers in other companies will not consult guides or use references, but rather enquire into the capabilities of the production facilities and sub-suppliers respecting little more than the laws of physics.

This is certainly the case at Grundfos TC, where product components are by default dedicated to only a single product. Even when subassemblies from earlier projects are reused in product development, they are often slightly modified and the information is only informally shared among designers and passed on downstream.

It is my belief, that Grundfos (and many other companies) can learn a lot from LEGO in this aspect and LEGO can learn a little from Grundfos as well (described in the following subsection).

The LEGO case is an exemplar on how to update platforms by adding new components, which comply with the rules of the platform. New platform components can be added by product developers, who will design the components in compliance with guides or references created earlier. These platform components are then added to the platform component assortment and made available for other projects. Alternatively new platform components can be added by platform developers or other employees dedicated to rationalization, but this will often be in the form of replacements.

Clean-up and replacements

The downside of the way LEGO designs new components is that *all* components are treated this way. All components are treated as if they are standardized platform components (see Figure 108). All components are added to the available component assortment. But not all components are meant to be shared widely across the product assortment – and indeed they are not.

Figure 108: At LEGO all components have traditionally been standardized and treated as evergreen. This has resulted in an ever-growing assortment and a slow devaluation of sharing, because no components were discontinued.

Because of the swelling of available components in the component assortment at LEGO, the company has had to undertake several clean-up processes in the later years. These are management-driven reactions caused by the constant swelling (see Figure 109) (Platform projects can in this aspect be considered an advanced clean-up project, which furthermore adds new components to the assortment). The clean-up processes have removed the least used components several times, which have largely been beneficial, but sometimes components, which should *not* be removed, are also removed.

The problem is that when components are removed because of their limited application in the current or past assortment, some components which could be applied extensively in the future assortment are also removed. Some themes and styles appear periodically in LEGO products, and even though some components are not used for several years, this does not mean that they will not be used again later.

This is however not a big problem. On a large scale the clean-up projects have been successful, and most product designers at LEGO agree that the removed components were the least useful. The remaining problem is that the clean-up projects so far have been reactive, based on the poor results and too high component costs. If results become better and less focus is being put on cost reductions, the component assortment is likely to swell again.

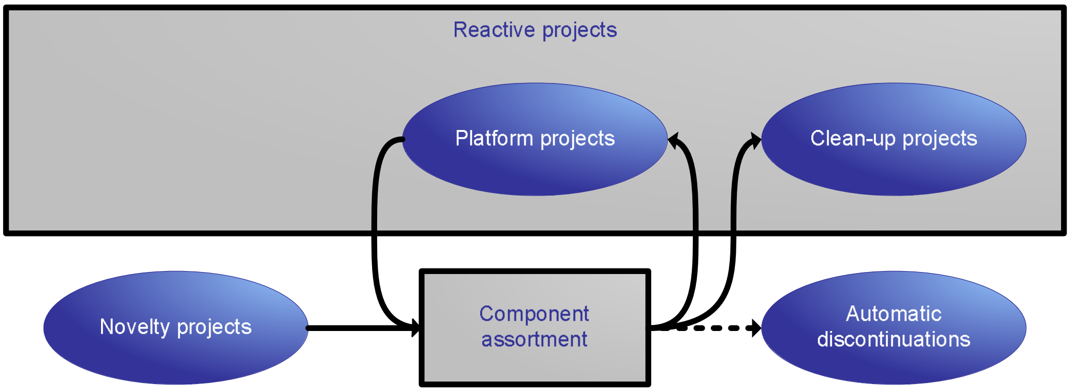


Figure 109: To reduce the component assortment two kinds of projects have been introduced: Platform projects and clean-up projects - both are reactive though.

Two new approaches for managing the size of the component assortment were introduced in the Minifigure Accessories project and have already been partially described in section 7.1.1.

The first approach is replacement. When each product designer group is given responsibility for the components, for which they are the primary users, they can also be given a frame for the number of components. For instance the Castle product group could be given a fixed frame of 30 minifigure accessory components. They would then have to replace components instead of adding to the assortment, and they could be given a fine equal to the value of any unreturned investments, if they replaced a mold before the investments were fully returned. The designers should then not be subjected to the existing annual component frames, but be able to replace as many components as they wanted, if they were able to pay the price of new molds and fines for discarding old ones.

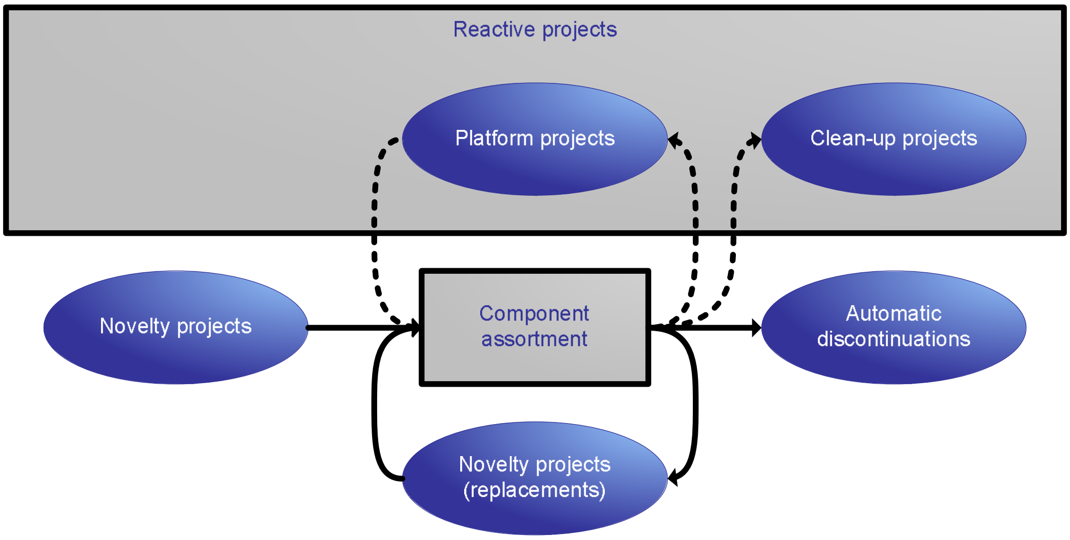


Figure 110: Suggestions from the Minifigure Accessories project. Novelty project teams can be made responsible for parts of the assortment and can therefore replace rather than add components, and unused components can be automatically discontinued. This will reduce the need for reactive clean-up and platform projects.

The second approach is an automation of the clean-up procedures. Some (but not all) components could be given a fixed time frame for lying idle (e.g. two years). These components would then be automatically put up for termination, when this fixed time frame had elapsed. An unfortunate consequence of this procedure is, as mentioned before, that some useful components will sometimes be discarded. Before this consequence is decided to be too grave, we should however consider the likelihood of product designers reusing a several year-old component (i.e. they will likely want to improve it), and we should also consider the costs of stockpiling and maintaining molds for extended periods of time (i.e. is it even going to be economically sound, and will it still work?).

The two approaches are shown in Figure 110. By empowering the novelty teams with limited responsibility of the component assortment and introducing automatic discontinuation of unused components, replacement and removal of components becomes more integrated in the annual novelty project process, and the need for reactive clean-up and platform projects will decrease and ensure a more stable and continuous balancing of the component assortment.

7.1.3. Monitoring platform performance

As mentioned in section 7.1.1, platform performance can from one perspective be measured by how much the standardized components are being reused. Every time a standardized component is applied in an additional product, the costs and time of developing an equivalent component has been saved, and in some cases the product would not have possible without the standardized (i.e. less expensive) component.

Consequently platform performance can be evaluated based on the utility of standardized components (i.e. how widely they have been applied) and assessments of the savings of utilizing the standardized component compared with utilizing a unique or modified component.

Platform cases with few standardized product parts, but with standardized architectures of generic placeholder modules, can be evaluated based on the utility of these architectures and the savings related to utilizing them instead of unique or modified architectures.

Assessing the savings of utilizing a standardized component instead of a unique or modified one can be surprisingly difficult and is by definition based on estimates and guesses (see section 2.6.1). But if we assume that the savings are both real and significant, then platform performance can roughly be evaluated using component utility alone.

Grundfos TC is evaluating platform performance based on this reasoning. If individual platform components have been shown to be beneficial in product development, then using these more frequently is good and using them less is bad. Currently Grundfos TC management is monitoring the usage of the standardized collapsible core units, and product developer performance evaluation is influenced by successful use of these standardized components.

Another aspect of platform performance is the number of standardized components. Is it growing? Or declining?

When standardizing a component assortment, we naturally want to increase the number of standardized components compared to the number of unique components, but the total number of components should decline (see Figure 111). Special platform projects will add standardized components and replace unique components.

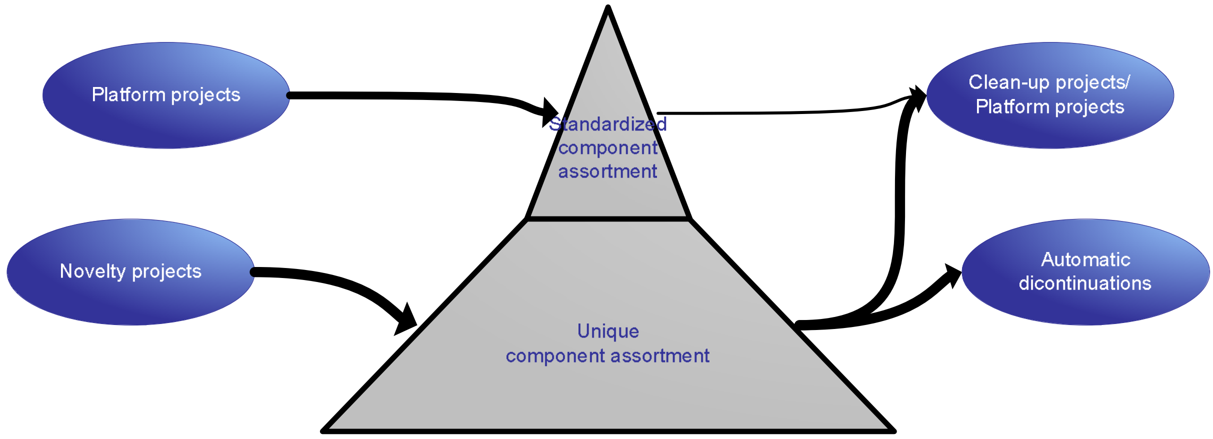


Figure 111: When standardizing a component assortment, the number of standardized components should increase, the number of unique components should however decrease faster, so that the total number decreases.

Over time however, if we assume that the non-value-added variety has largely been removed, the number of standardized components should stagnate unless we are also growing the production volumes (see Figure 112).

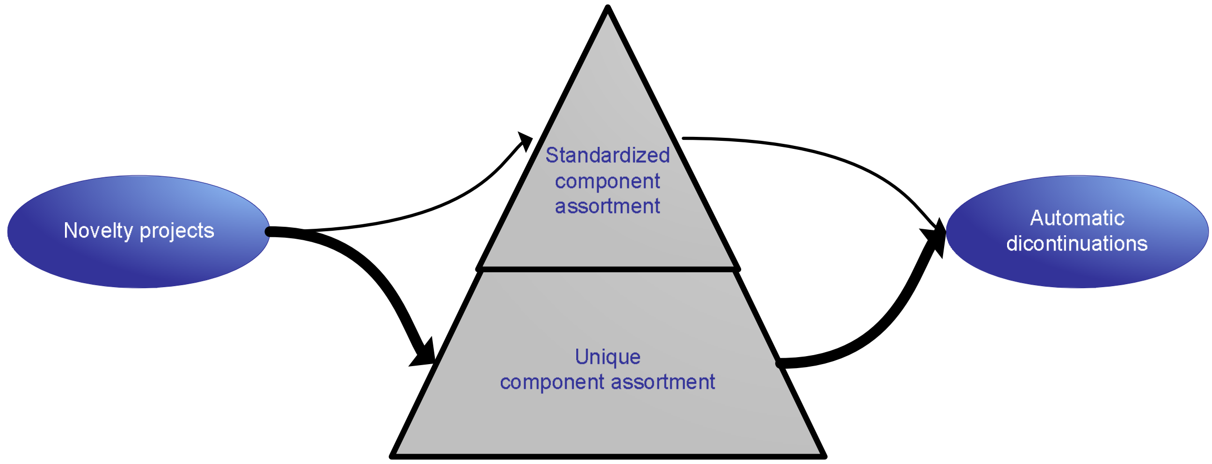


Figure 112: In the long run the component assortment should stabilize, which means that components should generally only be replaced. Unique components should naturally be replaced much more frequently, and therefore many more new unique components will be developed every year.

At LEGO, the general perception is that components have already been standardized to a very high degree, and the amount of non-value-added variety has been minimized. If this perception holds true, then LEGO will only need more standardized components if it grows the entire business.

Since standardized components (i.e. universal and generic bricks) are though to be evergreen, then we need no new standardized components, but since more unique components (i.e. special bricks) will only last for one or two seasons, we will constantly need to replace these. This leads to a situation where all new components designed every year are unique and very special – but this is no problem. What does matter is that roughly the same number of unique components should be removed every year, and the existing standardized components should be used extensively.

Improving more special bricks, so that they fit the criteria for generic or universal classifications will not increase the overall usage or production volume of components (based on the logic that using one brick more will naturally mean using another one less), and therefore doing this will slowly deteriorate the value of the UGS-classification.

The growth or decline of components compared to the production quantities can therefore be used as an indication of platform performance. When the production quantities are stable, a growth in standardized components should lead to an even bigger decline in unique components, if this is not realistic, then the number of standardized components should not grow.

7.1.4. Conclusion

Based on the cases, I conclude that platforms performance can be evaluated partly by how and how much standardized components are used, which correspond to the hypothesis, which I gave in section 3.3:

Platform performance can be evaluated based on how and how much the platform elements are applied in the product development projects.

My findings however show that this is only one aspect of platform performance. An even broader way of evaluating platform performance is by looking at the growth and decline of standardized and unique components and how this relates to the production volumes. Good-performing platforms will only experience growth in the number of standardized components, when production volumes are increasing or when the number of unique components is declining rapidly.

The other research question presented in section 3.3 about revisions and additions have also been examined in the previous sub-section, and the hypothesis, which I gave in the beginning, complies with the way additions and revisions are carried out at LEGO:

Specific rules and guidelines can be designed, which will allow additions and revisions to a platform, while ensuring the platform foundation.

It is obvious that this works perfectly at LEGO, since the components are reused and shared extensively. The specific rules and guidelines clearly defines how new components can be added to the component assortment, and many of the later initiatives (e.g. the component frames, the component costs structure, and the UGS-classification) aim at ensuring the foundation of the platform.

It should however be emphasized again, that continuously adding standardized components to a platform will eventually undermine the foundation. Therefore additions should be handled carefully and replacements should be preferred.

8. Synchronizing Platform and Product Development

This section aims at answering the research question stated in section 3.4 or confirming the hypothesis from the same section:

How can platform and product development activities be synchronized?

The question assumes that platform and product development are distinct activities, which is generally the case in the industry as observed by [Elgård 1998] and supported by the industrial cases in this thesis.

All state-of-the-art literature deals implicitly with interplay between product development projects and platform development projects, since the whole purpose of platform development projects is to create something which is useful in product development projects.

[Mortensen et al. 2005] describes advanced platform development includes a business process, where product development is aligned with product architectures, and where product development is a coordinated effort of parallel development of features, products, standard designs, architectures, and technology. These statements are however not detailed, and have therefore served as an inspiration in my research.

As stated before, the vast majority of papers and books however describe platform projects as special projects designed to make a permanent change in the products and in product development (at least until a new project changes things again), since they describe how to create platform components or architectures from scratch. This does not suggest a continuous process, and therefore most literature does not describe problems (like continuous synchronization) related to continuous platform development.

Almost all papers and books referenced in this thesis imply that the company first develops a product platform and then develops a range of products based on this. The platform is in some cases described as being based on an existing product family.

But how does this relate to new product development? How can we apply platforms in an innovative product development organization?

These questions have already been answered partly in section 6.1.2. In this section it was described – based on [Sanderson et al. 1995, Elgård 1998, Martin 1999] – how platform approaches are more applicable in the later stages of a product s-curve. In this section however, I will go into the details of platform and product development interplay in these later stages.

If we perceive product and platform development as continuous processes which run in parallel or possibly alternate then there is no clear beginning or end. In this mindset the product development process continuously aims at developing specific products, while the platform development process continuously aims at developing generic things, which may be utilized in the near-future product development process.

There are two primary challenges here. First, we need to find out how to divide the tasks and how to handle them separately. Secondly, we need to find out how to bring the developed components back together.

8.1. Extracting platform activities from product development projects

An important part of synchronizing platform and product development activities is about planning what to develop as dedicated special components aiming at specific appliances and what to develop as common generic components intended for sharing and reuse in several projects.

8.1.1. Generic and dedicated components

Classifying components as either generic or dedicated should be based on the principles of value-added and non-value-added variety described in section 2.4.1. To summarize the findings of that section, products should vary on a few important features, and the remaining product components should be identical.

The identical (or recurring) components should be designed as standardized components and designed in a way which makes them relative independent of the design of the various new features which will provide the unique value for each product variant. This is what constitutes the platform design activities.

The unique features of the product variants should be designed as part of the regular product design activities.

The classification and division itself should be done upfront whenever a new product is put into the pipeline – at least to the extent possible.

8.1.2. Benefits of having two separate activities

The Grundfos TC molding equipment platform project (from section 4.2.1) showed how it was possible to separate the product development process into a generic platform design process and a dedicated product design process (i.e. a preparation stage and an execution stage).

These two processes runs in parallel and the main benefit of this separation is of course the already mentioned reduced lead-time. Theoretically rearranging two sequential activities into two parallel activities would reduce the lead-time only if sufficient capacity was available. But even though many companies would reluctantly increase their capacity if shorter lead-time could be achieved, this is not necessarily required.

Due to the different time-spans of product development activities, the various capacities of a product development company will be differently stressed. Product ideas will not appear at fixed intervals and even if they did, concept design and product design will not always take the same amount of time. Because of this production and other downstream activities will not start at fixed intervals and there will naturally be times when the production is idle and times when it is over-burdened (see Figure 113).

The classical fix for this problem is to plan ahead and try to create fixed deadlines, so that downstream activities can also be planned. This however will often mean that some projects are rushed and some projects are given more time than required. Furthermore, even at LEGO where conceptualization is put into a strict timetable and all activities are given fixed deadlines due to the seasonal sales described previously small variations in project-handovers and differences in the complexity of the projects themselves create fluctuations in production.

A great benefit of separating the product development activities into preparation activities and execution activities is that the preparation activities are often more flexible (see Figure 114).

At Grundfos TC the generic mold parts can be produced at any time from a new order arrives, whereas beforehand all production would wait until the design was done. Similar ideas have been proposed by different employees at LEGO, and a major driver of the platform projects has been to be able to produce these bricks before the high-season starts.

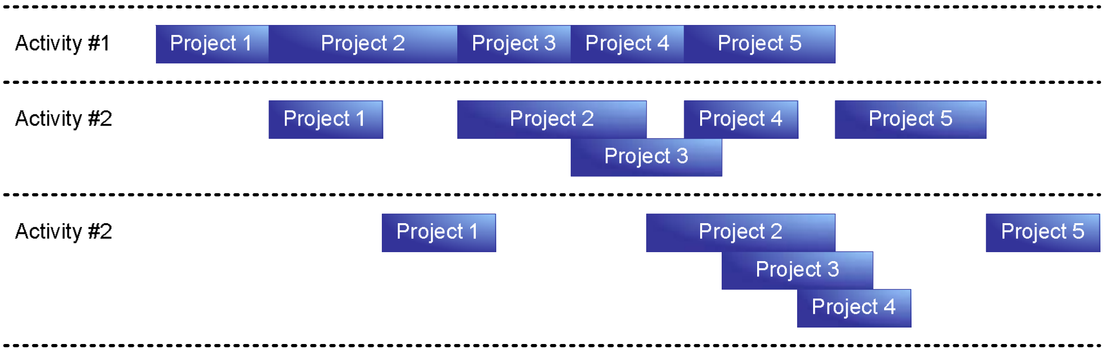


Figure 113: Variations in the time-span of activities create increasingly bigger fluctuations in the stress of capacities downstream, when these activities are sequential. The final activities experience idle-time and times of great stress.

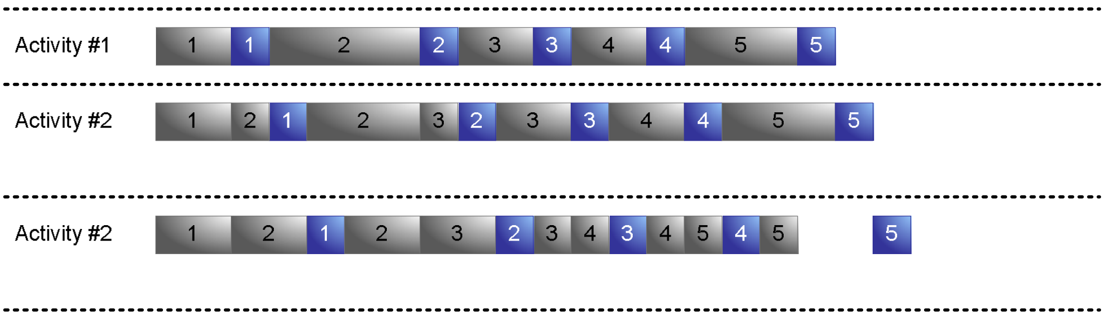


Figure 114: Example in Figure 113 separated into preparation activities (in gray) and execution activities (in blue). The execution activities are sequential (i.e. the previous activity must have ended before the next can begin). Preparation activities are more flexible because they do not require a previous activity to have ended before they can be initiated, and therefore this Figure shows less fluctuation in the stress levels.

It is the impression in my research group that using preparation activities to even out design and production fluctuations in this way could be useful for a broad range of companies.

8.2. Bringing it back together

When product development has been separated into preparation activities and execution activities it is naturally important that these two processes can be brought back together. But this is not always simple.

8.2.1. Aligning platform design and product design

A major challenge of running two separate product development processes which must later be joined is to ensure alignment so that the two processes do not go in different directions.

It is essential that the platform design team continuously learns of the new products, so that the platforms are prepared for these exact products. Similarly the product design teams must continuously learn of the platforms, so that new platform components may be exploited and implemented.

This challenge was recognized early on at LEGO and the platform development plans in the Screening 2005 project showed regular milestones in the development process, where the two processes would be aligned (see Figure 115).

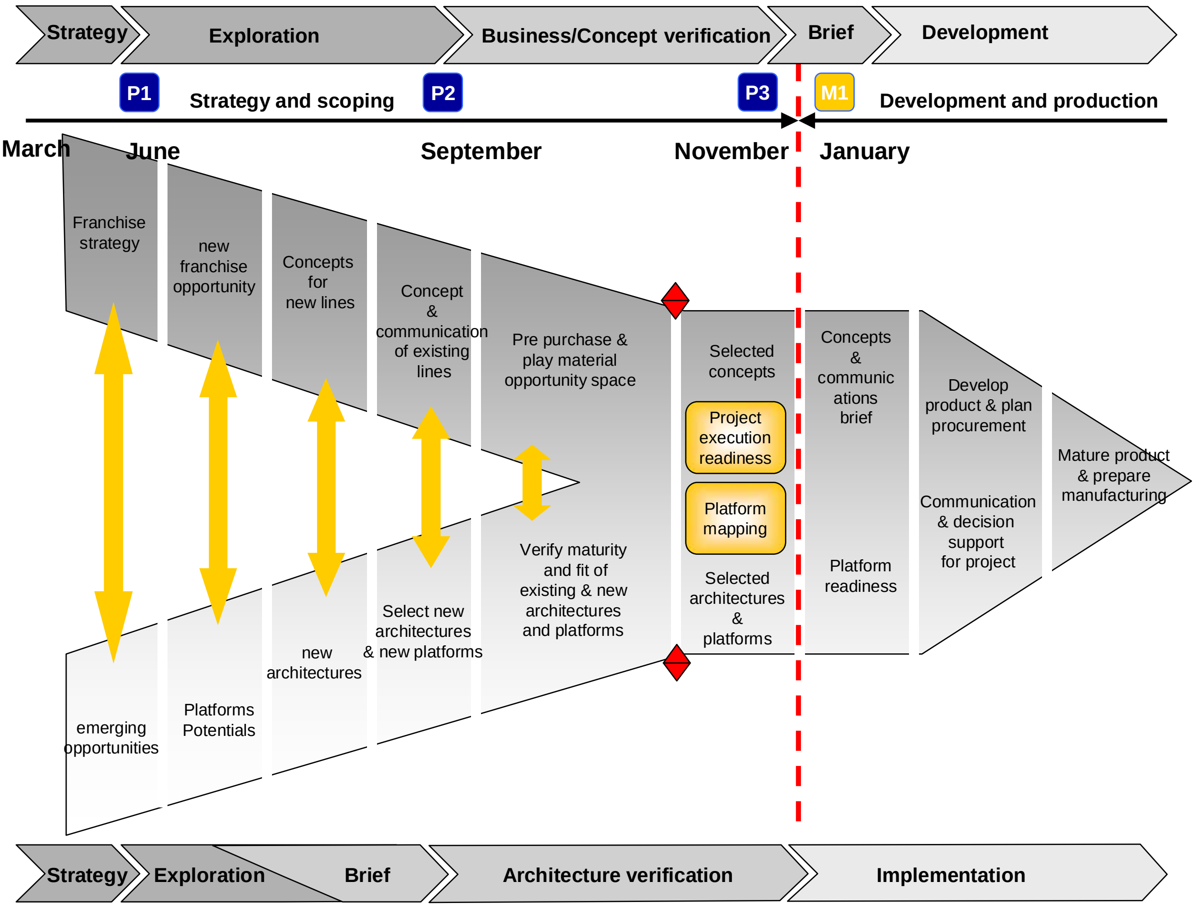


Figure 115: Bringing the platform design process and the product design process together at LEGO, fall 2005. Before the two processes are brought together they are continuously aligned. Some slides from that fall showed that management (Commitment Forum) was responsible for this alignment.

The failure of the Screening 2005 project (described in section 4.1.3) meant that these plans and milestones for synchronizing and aligning the processes were never implemented, and the following year all focus was on showing results and getting platforms through rather than on formalizing the procedure. The general ideas were however not discarded and the principle of regular milestones with alignment between the two processes was followed although the alignment fully became the responsibility of the platform development team.

8.2.2. Finding a balance between platform development time and platform irrelevance

Not only must the two processes go in the same direction, they must also meet at the same time. Platform components need to be ready when they are needed.

The platform projects at LEGO have unfortunately highlighted this all too well. Especially in the first year of the Universal Elements platform project, the plans for the platform components and the plans for the other components collided. The platform components were simply not completely done when they were needed and they also used up a part of the much-needed high-season production capacity. This was partly due to a too ambitious development and launch plan for the platform components although a number of other factors related to the concurrent out-sourcing of brick production had also influenced the plan-collision. The conclusion was that the activities had begun too late.

The following year at LEGO the platform projects got a new development and launch plan, which was moved a month backward compared to the previous year. It quickly became apparent however that this was not carried out so easily.

The new product ideas were not mature enough for us to be able to extract platform component candidates, and even though the plans of the previous year had collided, the designers and managers were reluctant to start the planning of platform components earlier. They feared that platform components derived so early would not match the actual need later on and therefore be irrelevant.

This must be a general observation. Platform components developed late will miss the product launch or interfere with the execution activities. Platform components developed early may not be adequately specified and therefore be irrelevant or reduce the quality, when the product is finally done (see Figure 116).

Figure 116: Late platform development risks collision with the execution product development activities and early platform development risks irrelevance due to unfinished specifications.

The result at LEGO was that the platform design was rushed, but this is of course a fine balance to master. Rushing the projects too much will endanger the quality, whereas giving the projects too much time will result in the outcomes described in the above.

In the Grundfos TC test bed platform project great effort was also put into merging the platform project with upcoming product launches. Platform components needed in these product launches were rushed and the platform concept was aligned with the needs of the product launches.

In the researched cases platform projects were in general rushed to meet the deadlines of product launches and but it cannot be concluded that product launches never are postponed until a platform is done. In the case companies described in this thesis however, product launches carries far more importance than platforms and the platforms must therefore always follow the product pipeline.

It is of course not always possible to rush a platform project (or doing so would be unwise), and therefore several platform projects at LEGO have been postponed until a following year. Some platform projects however cannot be undertaken within one single year because they require a more in-depth analysis and possibly experiments. These projects are currently not being initiated at LEGO because they cannot be completed in time for any upcoming products and therefore given lower priority compared to shorter platform projects.

8.2.3. Implementation of platform components in products

It is not enough to align the plans and merge the projects at the right moment. In section 6.2.2, I listed a number of typical causes for platform project failure observed at LEGO. Among these were a bad implementation plan and lacking commitment from the users. In section 6.2.2, I merely concluded that an implementation plan and a stakeholder-list should be in the general presentation of the platform project. In this section however, I will describe what to actually do in a project.

Transition plan

When we think of platform projects, which result in a number of new components for a range of new products, all is well and good. In reality these components should often also replace existing components in existing products – or this is at least the assumption in most platform business cases.

The fundamental reason for introducing generic components is that they should be shared between products, and while this sharing may be between a range of future products exclusively, it does most often also include present products.

At LEGO almost all of the platform components substitute existing components. Likewise at Grundfos TC platform components like the collapsible core or the various test bed components replace existing components. Because of the special nature of Grundfos TC, they only produce one or a few products based on a product design, and therefore replaced components, generally should not be replaced in existing product lines. This is however special for Grundfos TC. Most other companies which my research group has studied have product lines, which should be adapted to the new components.

It has been observed by many people from the industry, that even though platform projects should decrease the total number of components (see section 0), that for a transition period the number actually increases. This observation is of course based on the fact that old components cannot be removed instantly, but must be phased out gradually (see Figure 117).

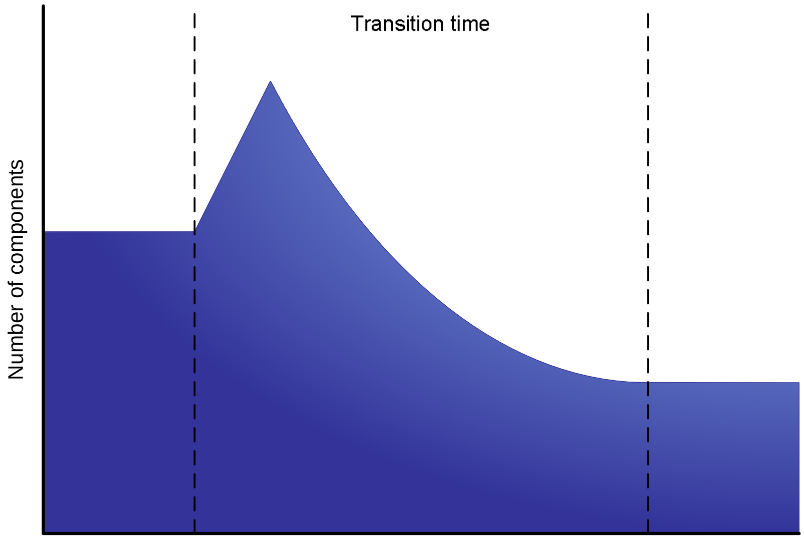


Figure 117: Component growth in the transition period. New components are added instantly (or near instantly), while old components are gradually phased out. The result is a period of even more components.

The Wheels platform project at LEGO highlighted the problems of this phenomenon. When the new platform components had been designed and were ready and available, old components should gradually be substituted. During the platform project it had been decided to extent the transition time. This meant that the peak of Figure 117 would be avoided and that investments would be spread out it also meant that the transition would not be complete until five years later.

During those following years the Wheels project were faced with numerous accusations. The individual substitutions were questioned and the project did not show the expected results, because the investments had not been brought down significantly and the assortment was still confusing, which was only natural during the transition period.

The main problem with the substitutions was that a chart showing the individual substitutions had never been created prior to the final milestone. Only charts of the new assortment had been presented, and when faced with the actual substitutions, product designers were reluctant to let go of the old components. Also, even though many designers approved of the new assortment on a longer scale, some of the new components could not directly substitute old ones in existing products.

An elaborate diagram showing the out-phasing procedure was created in the transitions period for the Wheels project, this diagram ensured that old components was constantly sought substituted and discontinued. Without such a diagram and the heavy focus on reductions at LEGO those days, a very real outcome could have been, that some of the old components would never be fully substituted (something which had occurred in previous Wheels project attempts).

When looking at platform project synchronization and implementation as a continuous process, it is clear that some of these issues should be addressed.

First and foremost, the temporary increased assortment and lack of immediate results should always be included in platform presentations. Results should always be split into long-term and short-term. Management and users must be told that the project is not over until the transition period has passed, and that all results will not appear until this happens.

Secondly, when new components should substitute old ones directly in products, it is important to show this explicitly as a part of the visual platform concept, and to find out if these substitutions are likely to be successful.

Creating commitment

To prevent failure when bringing the activities back together, it is also necessary to gain commitment. Getting the necessary commitment from the product developers does not come natural. Platform projects are likely to impose restrictions for designers in the form of design rules and guidelines which is likely to meet opposition in any organization.

Involving product designers in the platform development process helps to determine the requirements of the product platform, but it also helps to create acceptance of the whole platform project. Creating acceptance or even approval is important in a platform project, because the effects of the platform will be greatly diminished if the behaviour of designers remains unchanged (i.e. as if no platform process existed). Behavioral change can be facilitated by procedural change, but these procedures could be short-lived if they are not approved of and enforced.

My research offers no general applicable tools for creating platform approval and commitment in an organization, since it has not been the direct object of any case studies. It has however been an element of nearly every mentioned project and I have made a number of useful observations.

One of the biggest threats to the Wheels platform project was the lack of product designer commitment. If the designers continued to introduce new wheel components as if the platform did not exist there would be no beneficial outcome of the platform.

By achieving the short term goal of the project (i.e. to meet a breakeven situation fast) a commitment from management could be achieved. This goal was obtained by reducing investments by rebuilding existing moulds, postponing individual investments until needed, and accepting some bad elements for a transition period. Furthermore a number of economical scenarios were created which gave predictions of the outcome of the project in various situations and which offered management greater influence on the implementation. Finally, management gave it's commitment to the project.

To enforce this commitment from management new procedures for using wheel components in the products were implemented. These procedures made it possible to monitor the designers' use of unwanted wheels in new models. The use of any non-platform wheel components would not be impossible, but sound argumentation would be required to do so. A scale showing the effects of different designer actions and the likely reactions from management was introduced to explain the design rules of the platform.

To further explain the new platform and create a general approval of the new system a number of presentations were made for the designers, and among other initiatives a short story (see Figure 118 – exact wording is confidential, therefore it has been downsized) was created. The short story explained the reasoning behind the new platform.

Figure 118: Fictive short story from the Wheels platform project at LEGO. In this story the fictive main character, Jørgen, comes up with the Wheels platform concept. The exact wording is confidential. The story was devised as part of the project in order to communicate the reasoning behind the project and platform projects in general to all employees and received great attention.

Even though a number of specific and general presentations of the Wheels platform project had been given, product designers did not approve of many specific decisions taken in the project. The project did turn out successful, but this was primarily a result of management commitment and the fact that the project team remained active in the extended transitions period. The main reason for the lack of designer approval was that they generally did not feel that they had been consulted when the various important decisions where made.

The product designers had actually been invited to participate in the process, but most had declined the invitation, because they did not really know what platform projects were all about, and because they had other important things to do. This was of course unfortunate, but it seems an almost unavoidable event, when a new largely unknown process is introduced.

In the following platform projects we put even more effort into engaging the product designers, but the product designers themselves had also learned from the experience and were much more willing to participate. The result has been that some product designers have taken an active role in the projects and that these designers therefore have been promoting the platform concepts themselves in the organization.

In Grundfos TC some of the lead designers were involved from the very beginning. This is a big benefit of anchoring the platform projects in the product development teams directly. The lead designers influenced the platform projects and promoted the idea in their organization. In Grundfos TC the problem lies more in initiating new platform projects and evaluating the older ones – something which LEGO excels in.

It seems in general that a combination of getting management commitment to enforce the procedures and decisions of the project together with product designer involvement and communication of findings and results on a broad scale are some of the elements which can be used to ensure commitment for the platform projects.

8.3. Synchronization conclusion

Synchronization of platform development and product development is the art of separating the two processes, aligning them although they are separate, and bringing them back together.

In my hypothesis, I dealt primarily with how the two processes were brought back together:

When a company develops platforms continuously, the platform projects runs in parallel with the regular product development projects, adjusted so that any platform project is always slightly ahead of the product development projects, which builds upon that platform.

When thoroughly exploring and examining how to synchronize the processes, I have found that this answer is lacking. Not only, does it not answer why platform projects must run slightly ahead of product development projects, it also does not answer how the two processes run in parallel, how direction and timing are aligned, and how good implementation is ensured afterwards.

In the LEGO cases the processes are continuously aligned with each other to ensure the direction of platform project and the timing. The continuous alignment in the form of presentations and product designer involvement, not only ensures that the platform project delivers things which are useful, but it also helps on product designer approval.

Concerning the timing of the platform project I have presented a balance between getting the platform ready and available in time (i.e. before the product development project ends) and beginning the platform project when sufficient information is available (i.e. after the product development project starts) – and still having the time to complete the project.

Finally I have found that the transition period, when the platform is being implemented, is important, and that scenarios describing this period must be created as part of the platform project, so that nobody is surprised when component quantities rise after a platform project is done.

9. Conclusion

In this thesis I have presented my findings and results from my studies of the literature and of several industrial cases. The focus has been on continuous product development, which is the name I have given to the ongoing process of developing and managing product platforms.

Although most publications state directly or imply that platform development is not a one-time event, no research has focused on the continuous aspect. The findings presented here indicate that the challenges of continuous platform development are not simply the same as for traditional platform development, but that new challenges arise with this concept.

Fortunately I have had the opportunity to study some of these challenges and how to deal with them at LEGO Company, where platform development has been around for so long, that it is becoming a continuous process.

In its totality this thesis presents a new view on platform development, where new platforms or platform components are developed regularly in a structured process. In this structured process platform projects are suggested, prioritized, screened, and aligned with simultaneous product development projects before becoming true platform projects. Furthermore platform component use is monitored and adjusted by addition, replacement, and removal of platform components, which again lead to new platform components.

Some major benefits of traditional platform development like economies of scale are partly forfeited, investments in platform components and activities will rise, and a the responsibility of developing and managing platforms must be permanently anchored in the organization, all to ensure that the lead-time reduction remains and the platforms stays relevant.

Part-conclusions and answers for the various specific research questions presented in the beginning of this thesis have already been given in the previous conclusion sections 6.1.5, 6.3.3, 7.1.4, and 8.3. The purpose of this final conclusion is therefore to evaluate the already presented conclusions and to put them together in a context.

9.1. Scientific contributions

In this thesis I have posed and answered questions detailing areas of platform development, which has so far not been covered by state-of-the-art platform literature and therefore constitute my contribution to the research area.

1. I have described (in section 6.3.1) how the overall platform development scheme can be divided into three separate high-level stages: The idea generation, the project elimination race, and the true platform development stage. This was summarized in Figure 106.

This approach is new and although it has been hinted at by a few authors, it has never been thoroughly dealt with. The division itself is not remarkable, and a number of alternative separate steps could have been proposed. The contribution here is the process that the steps span: the idea generation, the elimination race, and the true platform development.

2. I have also described (in section 6.1) how good platform ideas can be found and qualified by consulting a broad range of company employees from marketing and sales, product development, and downstream departments combined with persons who have management insight, product insight, and platform insight. And that this is

only possible when the timing is right in the market and in the company. This is summarized in Figure 91.

This conclusion is maybe not very surprising, but it does highlight an area, which has not been covered before and it does suggest that the majority of companies, who expect product developers to be responsible for suggesting new platform projects to be too simple.

3. Next, I have described (in section 6.2 and 6.3) how platform projects can be screened and prioritized in comparison. And that this can be done in iterative shared presentations by presenting comparable costs and benefits, visual platform concepts, implementation plans, and commitment statements.

Although few companies are likely to launch platform projects in the way and at the rate that LEGO has done, the principles of iterative presentations and the standardization of presentation content is widely applicable. The suggested content of the presentations are compared to other platform presentations I have witnessed a valuable shift from a traditional quantitative presentation of benefits and costs to a more qualitative presentation of the platform concept. And again this is not something, which has been researched in state-of-the-art publications.

4. The next contribution is the model (in section 7.1.2) describing component additions and replacements, which shows that components must be taken out of the assortment or replaced on a regular basis in order to stabilize the assortment and prevent radical clean-up projects. This is summarized in Figure 110.

This conclusion deals with part proliferation, which has been described by many authors as a growing problem in product development companies. In this context however I describe it as a result of running platform development projects without an after-though on the component assortment. The model must also be seen in a platform development context, as it extends the responsibility of the product developers into not only adding but also removing and replacing platform components. It is implied in the state-of-the-art literature that standardization prevents part proliferation to some extend, but in my observations standardization may also lead to part proliferation because standardized components are kept.

5. The fifth contribution is the insight (in section 0) that overall platform performance can be evaluated based on how and how much platform components are being used and on the growth and decline of standardized and unique components.

This contribution is really an alternative approach to evaluating platforms, which may not be very accurate but which is easily applicable in many contexts. State-of-the-art literature present a few other approaches to evaluating platforms, which are more detailed and valid but also less applicable.

6. The final specific contribution is the description (in section 8 – especially 8.2) of how platform projects must be continuously aligned with product development projects and how it must not be started too early nor end too late compared to these product development projects in order to succeed.

This conclusion is not very surprising, except for the fact perhaps that platform projects must not be started too early. This is of course relative. Many companies (including Grundfos TC) perceive themselves as basically developing and producing the same thing, and for these companies can therefore specify the platforms as early as they like. The fact remains however that projects with too long a development plan will receive will loose attention in a company. And again this has not been covered by state-of-the-art literature.

Following the specific contributions, I would like to emphasize the overall presentation of the platform paradigm:

7. The initial description (in section 2) of the platform paradigm gives a coherent picture of the motivations for introducing platform-based product development as well as the characteristics and the effects of developing platforms. This is furthermore viewed in a new perspective (i.e. continuous platform development).

Although this description and the conclusions presented in section 2 are based on already published literature, common conceptions, and logical reasoning, they give a new and different overview of the research area, which in my mind qualifies as contribution to the research area.

8. Finally, the classification and description of platform projects (in section 5) and the following descriptions of platform screening, platform maintenance, and platform synchronization (in sections 6, 7, and 8) also extend the presented platform paradigm.

These sections put together are really the description of continuous platform development and what it is all about, something which is altogether new for the platform development literature.

9.2. Validating the results

The majority of the conclusions in this research build on a number of cases, where it has often been the case that an unwanted phenomenon has been observed in one or more platform projects, and steps to prevent the phenomenon has been taken in the following projects.

A major concern is however that many of the observations have not been confirmed in other companies, but merely build on experiences from LEGO. This applies for the first, the third, and the fourth contributions listed in the section above. Those three contributions rest primarily on observations at LEGO and have not been confirmed by observations at Grundfos TC.

The conclusions that I make and the suggested procedures and models are however sought generalized, and are far less detailed and specific than the original observations and models (some of these models can be seen in slides shown throughout the thesis). For five of the contributions this has meant that they have to some degree been confirmed by observations in the Grundfos TC cases, but this also means that they are relatively abstract and that they need to be detailed when applied in real cases.

Another concern is the relative high level that observations have been made on. As stated in the introduction this high level makes it harder to keep other factors equal, when trying to recreate observations, and many of the observations and cases presented in this thesis has also been accompanied with a number of alternative causes which may have triggered the observed phenomenon.

All in all, this means that the contributions have only been initially scientifically validated and that the models and insights could use further confirmation by being applied in more cases. As stated in the introduction, I believe however that this is a relatively common trait for contributions within this research area and that it is a condition for the research approach rather than anything else.

9.3. Implications of this research

Some of the conclusions and models presented in thesis have not been drawn prior to the writing of this document, and therefore have not been applied or presented in any of the case companies.

I will therefore use this opportunity to present the possible implications of this research in those two companies to demonstrate to full implementation of continuous product development.

9.3.1. LEGO Company

Most of the aspects of continuous platform development presented in this thesis are already being applied at LEGO. Platform projects are suggested, screened and prioritized on an annual basis, and the individual projects are synchronized with product development, so that they are available at the right time in the right form.

What remains, lies in my opinion primarily within platform and component maintenance. The handing-over of platform projects from the project team to the platform organization (i.e. DesignLab) is still not optimal, and the monitoring and maintenance of the component assortment is reactive rather than active as presented in section 7.1.

If platform developers at LEGO were to embrace the findings in this research and apply them, a number of procedures would change:

1. Product development teams would be given responsibility of parts of the more special and dedicated component assortments. It would hereafter be their responsibility to add, remove and replace components within this part of the assortment, and they would pay the costs of replacing components prematurely.
2. Special components would automatically be discontinued, when the products they were developed for stop being produced (unless another product could immediately use them). These components should already have returned their investments at that time.
3. The development of universal and generic components, which are not related to specific themes, should be taken completely out of the novelty teams. Any suggestions from the novelty teams should be fed directly into the platform development process.
4. An overall plan for the universal components should be created. This plan should detail how many more universal components are needed, so that a stabilization of the component assortment can be achieved (assuming that a continuous growth is unwanted). After the specified quantity of universal components has been reached, development of these components should all but cease.
5. Monitoring of the use of all components in products should be further streamlined, so that this information is readily available for product developers and managers.

A few more fundamental observations relates to component management although these have been the focus of this thesis.

6. While the UGS-classification is good, it is not perfect. As stated in the case descriptions it is based almost exclusively on geometrical properties, and therefore only indirectly ensures that the components are valuable to the customer, profitable, and can be developed and produced quickly at low risk. Because of this, the UGS-guidelines should perhaps be extended or replaced so that it classifies components correctly with greater accuracy.

7. Finally, the current UGS component costs reflect only the actual costs of producing the components as stated before. This can in a worst-case scenario lead to a situation where certain kinds of components always are preferable to the product designers on a short term, while other kinds of components never turn profitable. Management could however use true penalties or discounts to steer the development of the component assortment in a given direction aligned with the company strategy instead of leaving this to chance.

Having participated in LEGO projects for several years and discussed the component strategy with numerous company employees, I realize that these are not simple suggestions. The current procedures have been implemented with high costs and much effort, and new procedures which are perhaps more complex could lead to unwanted disturbance, which could ruin the results.

I will therefore suggest a gradual change.

9.3.2. Grundfos TC

Grundfos TC is in a more traditional situation than LEGO. Several platform projects have been undertaken, but the process is still far from continuous. As each project ends, the development of platforms within that area stops. Management and key employees in Grundfos TC however aims to improve the platform development within the department and make it an integral part of the product development.

Was Grundfos TC to embrace some of the concepts in this thesis the most obvious starting points would be the following:

1. To find and initiate new platform projects, the department needs to find more platform thinkers (they already have insightful experienced workers and managers, which are also needed) who can promote the platform paradigm, spark ideas, and facilitate the platform development. They could also further involve other departments to qualify platform suggestions and produce even more ideas.
2. Secondly, they need to properly differentiate platform components from dedicated components and create design rules, which would allow product developers who are not part of platform development to add to the platform concepts. In this they could learn from the way LEGO handles universal components.

The remaining concepts of this thesis are more relevant for Grundfos TC on a longer term. The department has however a good opportunity of achieving even more than LEGO, because they are not obligated to renewing their products for the sake of renewal alone in the same way as LEGO is.

9.4. Further research

As this research project can be classified as macro-research, there are naturally possibilities for further research within most of the contribution-areas that I have covered. Some areas are however more obvious than others.

In the area of platform maintenance, I have only studied component management. Some of the concepts and findings can perhaps also be applied for architectures in applications where these are more important than components. Architecture management and maintenance is therefore an obvious choice for further research.

A related area is the platform architects themselves, which have been discussed briefly in this thesis. Research that focuses on these persons and their responsibilities would clearly benefit the continuous platform development perspective. It is assumed in this thesis that architects have other responsibilities than the development of new platforms, but an extensive study of these responsibilities has not been carried out.

Two areas in the very end of this thesis could also need some more dedicated research. One of these is the transition period and implementation. I have deliberately focused on the earlier stages of platform development projects and therefore this area is not described in great detail. As stated in the sections about platform presentations and later related to re-joining platform and product development activities, this is however a critical period for platform projects, because the details of the period is often not given much thought in the earlier stages of platform development.

The other area which could need more dedicated research is the commitment creation, which is essential in many projects, but here the observations from this research project is insufficient to make more detailed conclusions than what is presented in this thesis. Such a study would also need to refer more to the change management literature than is the case here – this would however be a study worth of a PhD project in itself.

10. References

[1] Andreasen, M. M., Mortensen, N. H, and Harlou, U. (2004): “Multi Product Development: New models and concepts”, Proceedings of the 15th Symposium “Design for X”. Neukirchen, Germany, 2004.

[2] Andreasen, M. M., McAloone, T. C., and Mortensen, N. H. (2001): “Multi-Product Development – platforms and modularization”, The P* Programme’s series of Insight Reports, Technical University of Denmark. Denmark, 2001.

[3] Arbnor, I. and Bjerke, B. (1997): “Methodlogy for Creating Business Knowledge”, 2nd Ed. Sage Publications, Inc., 1997, ISBN 0-7619-0449-2

[4] Baldwin, C. Y. and Clark, K. B. (1997): “Managing in an age of modularity”, Harvard Business Review, September-October 1997, 84-93.

[5] Berglund, F. and Claesson, A. (2005): “Utilising the concept of a design’s bandwidth to achieve product platform effectiveness”, Proceedings of the International Conference on Engineering Design ICED 05, August 15-18, 2005, Melbourne, Australia.

[6] Blackenfelt, M. (2001): “Managing complexity by product modularization”, Doctoral thesis, Royal Institute of Technology, Sweden 2001.

[7] Blessing, L. and Chakrabarti, A. (2002): “DRM: A Design Research Methodology”, Proceedings of Les Sciences de la Conception, March 15-16 2002, INSA de Lyon. Lyon, France, 2002.

[8] Claesson, A. (2006): “A configuration component framework supporting platform-based product development”, Ph.D. Thesis, Division of Product and Production Development, Chalmers University of Technology, Gothenburg, Sweden, ISBN 91-7291-791-1.

[9] Clausen, J. P. (2007): “Innovation through modularization”, Master thesis, MMT, Aalborg universitetscenter, 2007.

[10] Dahmus, J. B., Gonzalez-Zugasti, J. P., and Otto, K. N. (2001):“Modular product architecture”, Design Studies, Vol. 22, 2001, 409-424.

[11] Desai, P., Kekre, S., Radhakrishnan, S., and Srinivasan, K. (2001): “Product differentiation and commonality in design: balancing revenue and cost drivers”, Management Science, Vol-47 No-1, 37-51.

[12] Elgård, P. (1998): “Industrial Practice with Product Platforms in USA”, Proceedings of the 4th WDK Workshop on Product Structuring, October 22-23 1998, Delft University of Technology, Delft, the Netherlands.

[13] Erens, F. J. (1996): “The synthesis of variety: developing product families”, Ph.D. Thesis, Eindhoven University of Technology, Eindhoven, the Netherlands 1996.

[14] Erixon, G. (1998): “Modular function deployment – a method for product modularization”, The Royal Institute of Technology. Sweden, 1998.

[15] Fabricius, F. (1994): “Design for manufacture – guide for improving the manufacturability of industrial products”, EUREKA, IPU – Institute for Product Development, Copenhagen, Denmark, 1994.

[16] Farrell, R. S. and Simpson, T. W. (2003): “Product platform design to improve commonality in custom products”, Journal of Intelligent Manufacturing, Vol. 14, 541-556.

[17] Fiil-Nielsen, O. and Mortensen, N. H. (2006): “Implementing product platforms: a case study”, Proceedings of the International Design Conference – Design 2006, Dubrovnik, Croatia, 2006.

[18] Fisher, M., Ramdas, K., and Ulrich, K. (1999): “Component sharing in the management of product variety: a study of automotive braking systems”, Management Science, Vol. 45, No. 3, 297-315.

[19] Fiore, C. (2005): “Accelerated product development: combining lean and six sigma for peak performance”, Productivity Press, 2005, ISBN 1-56327-310-1.

[20] Gershenson, J. K., Prasad, G. J, and Zhang, Y. (2003): “Product modularity: definitions and benefits”, Journal of Engineering Design, Vol. 14, No. 3, 295-313.

[21] Gershenson, J. K., Prasad, G. J, and Zhang, Y. (2004): “Product modularity: measures and design methods”, Journal of Engineering Design, Vol. 15, No. 1, 33-51.

[22] Halman, J. I. M., Hofer, A. P., and van Vuuren, W. (2003): “Platform-driven development of product families: linking theory with practice”, Journal of Product Innovation Management, Vol. 20, 149-162.

[23] Harders, A. K., Heimdal, E. J., Markworth, S., and Ingvarsdóttir, T. M. (2008): “Implementering af modularisering hos LEGO – forudsætninger og påvirkninger” (in Danish only), Course project assignment, 41080 Design Methods, Technical University of Denmark, 2008.

[24] Harlou, U. (2006): “Developing product families based on architectures – Contribution to a theory of product families”, Ph.D. Thesis, Department of Mechanical Engineering, Technical University of Denmark. Denmark, 2006.

[25] Hiatt, J. M. (2006): “ADKAR: a model for change in business, government and our community”, Prosci Reseach, 2006, ISBN 978-1-930-88550-9.

[26] Hildre, H. P. (1996): “Mastering Product Variety”, Proceedings of the 2nd WDK Workshop on Product Structuring, June 3-4 1996, Delft University of Technology, Delft, the Netherlands.

[27] Hölttä, K., Tang, V., and Seering, W. P. (2003): “Modularizing product architectures using dendrograms”, Proceedings of the International Conference on Engineering Design, Stockholm, Sweden, 2003.

[28] Hvam, L., Mortensen, N. H., and Riis, J. (2008): “Product Customization”, Springer-Verlag Berlin Heidelberg, 2008. ISBN 978-3-540-71448-4.

[29] Kaplan, R. S. and Cooper, R. (1998): “Cost and effect: using integrated cost systems to drive profitability and performance”, Haward Business School Press. Boston, Massachusetts, 1998. ISBN 0-87584-788-9.

[30] Kotter, J. (1996): “Leading change”, Haward Business Review, Cambridge, Massachusetts, 1996.

[31] Kotter, J. (2006): “Our iceberg is melting: changing and succeeding under any conditions”, St. Martin’s Press, 2006, ISBN 978-0-312-36198-3.

[32] Kristjansson, A. H. (2005): “Platform assessment matrix (pamatrix): a method to support decision making on product platforms”, Doctoral Dissertation, Department of Engineering Design and Materials, Norwegian University of Science and Technology, Trondheim, Norway, ISBN 82-471-6945-2.

[33] Kuhn, T. S. (1996): “The Structure of Scientific Revolutions”, 3rd Ed. Chicago and London, University of Chicago Press, 1996. ISBN 0-226-45808-3.

[34] Kvist, M. (2009): “Product Family Assesment”, Ph.D. Thesis, Department of Management Engineering, Technical University of Denmark. Denmark, 2009. Working document.

[35] Liedholm, U. (1998): “Conceptual design of products and product families”, Proceedings of the 4th WDK Workshop on Product Structuring, October 22-23 1998, Delft University of Technology, Delft, the Netherlands.

[36] Lyly-Yrjänäinen, J., Paranko, J., and Suomala, P. (2005): “Measuring component commonality: A franework for tackling complexity in product development”, Proceedings of the 12th International Product Development Management Conference, Copenhagen, Denmark, 2005.

[37] Martin, M. V. (1999): “Design for variety: a methodology for developing product platform architectures”, Ph.D. Thesis, Department of Mechanical Engineering, Stanford University, USA, 1999.

[38] Martin, M. V. and Ishii, K. (2002): “Design for variety: developing standardized and modularized product platform architectures”, Research in Engineering Design 13, 213-235.

[39] Meyer, M. H. (1997): “Revitalize your product lines through continuous platform renewal”, Research Technology Management, Vol. 40 Issue 2, 17-29.

[40] Meyer, M. H. and Lehnerd, A. P. (1997): “The power of product platforms: building value and cost leadership”, The Free Press, 1997. ISBN 0-684-82580-5.

[41] Miller, J. A. (1996): “Implementing activity-based management in daily operations”, John Wiley & Sons, Inc. USA, 1996. ISBN: 0-471-04003-7.

[42] Miller, T. D. (2000): “Modular engineering”, Ph.D. Thesis, Technical University of Denmark, Denmark 2000.

[43] Moore, W. L., Louviere, J. J., and Verma, R. (1999): “Using conjoint analysis to help design product platforms”, Journal of Innovation Management, Vol. 16, 27-39.

[44] Morgan, J. M. and Liker, J. K. (2006): “The Toyota Product Development System: Integrating People, Process, and Technology”, New York: Productivity Press, 2006, ISBN: 1-563-27282-2.

[45] Mortensen, N. H., Pedersen, R., Nielsen, O. F., Harlou, U., Bøgh, M. H., Høgh, P. T., and Hvam, L. (2008): “Implementing a product platform in 35 man-days: the visual thinking approach”, International Journal of Mass Customization, Vol. 2, Nos. 3/4, 240-263.

[46] Mortensen, N. H., Harlou, U., and Andreasen, M. M. (2005): “Identification of platform levels in product development”, Proceedings of the International Conference on Engineering Design ICED 05, August 15-18, 2005, Melbourne, Australia.

[47] Mortensen, N. H., Nielsen, O. F., and Jensen, T. S. (2009): “Product Platform Screening at LEGO”, to appear in the International Journal of Industrial Engineering”.

[48] Muffatto, M. (1999): “Introducing a platform strategy in product development”, International Journal of Production Economics, Vol. 60-61, 145-153.

[49] Munk, L. (2009): Working document, Ph.D. Thesis, Department of Management Engineering, Technical University of Denmark. Denmark, planned 2009.

[50] Olesen, J. (1992): “Concurrent Development in Manufacturing – based upon dispositional mechanisms”. PhD-Thesis. Institute of Engineering Design, Technical University of Denmark. Denmark, 1992.

[51] Otto, K. and Hölttä-Otto, K. (2007): "A multi-criteria assessment tool for screening preliminary product platform concepts", Journal of Intelligent Manufacturing, Vol. 18, 59-75.

[52] Patton, M. Q. (1987): "How to use qualitative methods in evaluation", Sage Publications, 1987, ISBN 978-0-803-93129-9.

[53] Pedersen, R. (2009): Working document, Ph.D. Thesis, Department of Management Engineering, Technical University of Denmark. Denmark, planned 2009.

[54] Ramdas, K. (2003): "Managing product variety: an integrative review and research directions", Production and Operations Management, Vol. 12, No. 1, Spring 2003, 79-101.

[55] Ramdas, K., Fisher, M., and Ulrich, K. (2003): "Managing variety for assembled products: modeling component systems sharing", Manufacturing & Service Operations Management, Vol. 5, No. 2, Spring 2003, 142-156.

[56] Riitahuhta, A., Pulkkinen, A., and Andreasen, M. M. (1998): "Metrics for supporting the use of modularisation in IPD", Proceedings of the 4th WDK Workshop on Product Structuring, Delft, the Netherlands, 1998.

[57] Robertson, D. and Ulrich, K. (1998): "Planning for product platforms", Sloan Management Review, Summer 1998, 19-31.

[58] Sahin, A., Terpenny, J. P., and Goff, R. M. (2006): "A graphical modeling environment for configuring modular product families and platforms", Proceedings of the ASME 2006 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, September 10-13, 2006, Philadelphia, Pennsylvania, USA.

[59] Sanchez, R. (2004): "Creating modular platforms for strategic flexibility", Design Management Review, Vol. 15, No. 1, 58-67.

[60] Sanchez, R. and Collins, R. P. (1999): "Competing in modular markets", 1999 Ron Sanchez and Robert P. Collins.

[61] Sanchez, R. and Collins, R. P. (2001): "Competing – and learning in modular markets", Long Range Planning, Vol. 34, 645-667.

[62] Sanderson, S. and Uzumeri, M. (1997): "The innovation imperitive – strategies for managing product models and families", Irwin (Times Mirror Higher Education Group) Inc. company, ISBN 0-7863-1009-X.

[63] Sanderson, S. and Uzumeri, M. (1995): "Managing product families: the case of the Sony Walkman", Research Policy, Vol. 24, 761-782.

[64] Sawhney, M. S. (1998): "Leveraged high-variety strategies: from portfolio thinking to platform thinking", Journal of the Academy of Marketing Science, Vol. 26, No. 1, 54-61.

[65] Simpson, T. W. (2004): "Product platform design and customization: status and promise", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol. 18, 3-20.

[66] Stone, R. B., Wood, K. L., and Crawford, R. H. (2000): "A heuristic method for identifying modules for product architectures", Design Studies, Vol. 21, 2000, 5-31.

[67] Thyssen, J., Israelsen, P., and Jørgensen, B. (2006): "Activity based costing as a method for assessing the economics of modularization – a case study and beyond", International Journal of Production Economics, Vol.103 Issue.1, 252-270.

[68] Törnebohm, H. (1974): "Paradigm i vetenskapens värld och i vetenskapsteorin", University of Gothenburg, Department of Theory of Science. Göteborg, Sweden, 1974.

[69] Ulrich, K. T. (2007): "Design: Creation of Artifacts in Society". Available at http://www.ulrichbook.org.

[70] Ulrich, K. (1995): "The role of product architecture in the manufacturing firm", Research Policy, Issue 24, 419-440.

[71] Ulrich, K. and Eppinger, S. (2000): "Product Design and Development", McGraw-Hill Higher Education. USA, 2000.

[72] Ulrich, K. and Tung, K. (1991): "Fundamentals of product modularity", Issues in Design Manufacture/Integration, ASME 1991, DE-Vol. 39, 73-79.

[73] van Veen, E. A. (1990): "Modeling product structures by generic bills-of-materials", Doctoral Thesis, TU Eindhoven, the Netherlands.

[74] Van Wie, M. J., Rajan, P., Campbell, M. I., Stone, R. B., and Wood, K. L. (2003): "Representing product architecture", Proceedings of ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Chicago, Illinois, 2003.

[75] Wadsworth, Y. (1998): "What is Participatory Action Research?", Action Research International, Paper 2. Available on-line: http//www.scu.edu.au/schools/gcm/ar/ari/p-ywadsworth98.html

[76] Wallace, K. M. and Ahmed, S. (2003): "How engineering designers obtain information", In *Human behaviour in design. Individuals, teams, tools,* edited by U. Lindemann, Springer-verlag, Munich, 184-194.

[77] Wilkinson, J. (1995): "Direct observation", in Breakwell, Hammond, and Fife-Schaw: "Research Methods in Psychology", Sage Publications Ltd, 1995, ISBN 0803977646.

[78] Yin, R. K. (1994): "Case study research: design and methods", Sage Publications Inc, 1994, ISBN 978-0-803-95663-6.

[79] Zamirowski, E. J. and Otto, K. N. (1999): "Identifying product portfolio architecture modularity using function and variety heuristics", Proceedings of the 11th International Conference on Design Theory and Methodology, 1999 ASME Design Engineering Technical Conferences, Las Vegas, Nevada, 1999.

Platform-based product development is about sharing components and product architectures between products in a product family to save costs and free engineering resources. The intension is that the components and product architectures must be shared both across existing and future products, but continuous product family evolution challenges this strategy.

The concept of continuous platform development is based on the fact that platform development should not be a one-time experience but rather an ongoing process of developing new platforms and updating existing ones, so that product family evolution does not make the platforms irrelevant. This study puts emphasis on platform projects and defines such projects as something very different from product development projects and more similar to industrial research projects, due to the fact that they are continuous activities with relatively low risks and investments but also with relatively fuzzy results.

ISBN 978-87-90855-88-8

DTU Management Engineering
Department of Management Engineering
Technical University of Denmark

Produktionstorvet
Building 424
DK-2800 Kongens Lyngby
Denmark
Tel. +45 45 25 48 00
Fax +45 45 93 34 35

www.man.dtu.dk